DALSA • 7075 Place Robert-Joncas, Suite 142 • St-Laurent, Quebec, H4M 2Z2 • Canada
http://www.imaging.com

X64-CL iPro™ Series

User's Manual

Edition 1.30

Part number OC-64EM-USER0

NOTICE

© 2009 DALSA Corp. All rights reserved.

This document may not be reproduced nor transmitted in any form or by any means, either electronic or mechanical, without the express written permission of DALSA Corp. Every effort is made to ensure the information in this manual is accurate and reliable. Use of the products described herein is understood to be at the user's risk. DALSA Corp. assumes no liability whatsoever for the use of the products detailed in this document and reserves the right to make changes in specifications at any time and without notice.

Microsoft is a registered trademark; Windows®, Windows® XP, Windows® Vista, and Windows® 7 are trademarks of Microsoft® Corporation.
All other trademarks or intellectual property mentioned herein belong to their respective owners.

Edition 1.30 released on: September 9, 2009

Document Number: OC-64EM-USER0

Printed in Canada

Contents

INTRODUCTION ____ **1**
Overview of the Manual 1
About the Manual 2
Using the Manual 2

OVERVIEW OF THE X64-CL IPRO SERIES ____ **3**
Product Part Numbers 3
About the X64-CL iPro Series of Frame Grabbers 5
X64-CL iPro Series Key Features 5
X64-CL iPro User Programmable Configurations 5
ACUPlus: Acquisition Control Unit 6
DTE: Intelligent Data Transfer Engine 6
Advanced Controls Overview 7
About the X-I/O Module 7
Development Software Overview 8
Sapera LT Library 8
Sapera Processing Library 8

INSTALLING THE X64-CL IPRO ____ **9**
Warning! (Grounding Instructions) 9
Upgrading Sapera or any DALSA Board Driver 9
Board Driver Upgrade Only 9
Sapera and Board Driver Upgrades 10
Sapera LT Library Installation 10
Installing X64-CL iPro Hardware and Driver 11
In a Windows XP, Windows Vista or Windows 7 System 11
X64-CL iPro Firmware Loader 12
Enabling the Camera Link Serial Control Port 14
COM Port Assignment 14
Setup Example with Windows HyperTerminal 15
Displaying X64-CL iPro Board Information 16
Device Manager – Board Information 16
Camera to Camera Link Connections 17
Configuring Sapera 19
Viewing Installed Sapera Servers 19
Increasing Contiguous Memory for Sapera Resources 19

Contiguous Memory for Sapera Messaging *20*
TROUBLESHOOTING INSTALLATION PROBLEMS 21
Recovering from a Firmware Update Error *21*
Windows Event Viewer *22*
Device Manager Program *22*
PCI Configuration *24*
Sapera and Hardware Windows Drivers *25*
Log Viewer *26*
Windows Device Manager *27*
Memory Requirements with Area Scan Acquisitions *27*
Symptoms: CamExpert Detects no Boards *28*
Symptoms: X64-CL iPro Does Not Grab *29*
Symptoms: Card grabs black *29*
Symptoms: Card acquisition bandwidth is less than expected *29*

CAMEXPERT QUICK START FOR THE X64-CL IPRO 31
INTERFACING CAMERAS WITH CAMEXPERT 31
CamExpert Demonstration and Test Tools *31*
CAMERA TYPES & FILES APPLICABLE TO THE X64-CL IPRO 32
CamExpert Memory Errors when Loading Camera Configuration Files *32*
Overview of Sapera Acquisition Parameter Files (.ccf or *.cca/*.cvi)* *32*
Camera Interfacing Check List *33*
LINESCAN EXAMPLE: INTERFACING THE DALSA PIRANHA2 LINESCAN CAMERA 34
CamExpert Interfacing Outline *34*
Step 1: Piranha2 in Free Run Exposure Mode *35*
File Selection & Grab Test *35*
Overview of Basic Timing Parameters *37*
Step 2: Piranha2 in External Exposure Mode *39*
Advanced Control Parameters *40*
Step 3: Piranha2 with Shaft Encoder Line Sync *42*
Shaft Encoder Line Sync Setup *42*
Shaft Encoder with Fixed Frame Buffer Setup *44*
Shaft Encoder with Variable Frame Buffer Setup *46*
USING THE FLAT FIELD CORRECTION TOOL 48
X64-CL iPro Flat Field Support *48*
Flat Field Correction Calibration Procedure *48*
Using Flat Field Correction *50*
USING THE BAYER FILTER TOOL 50
Bayer Filter White Balance Calibration Procedure *50*
Using the Bayer Filter *51*

SAPERA DEMO APPLICATIONS 53
GRAB DEMO OVERVIEW 53
Using the Grab Demo *53*
FLAT-FIELD DEMO OVERVIEW 54
Using the Flat Field Demo *54*

X64-CL IPRO REFERENCE **55**
X64-CL iPro Medium Block Diagram ... 55
X64-CL iPro Dual Base Block Diagram ... 56
X64-CL iPro\Lite Block Diagram ... 57
X64-CL iPro Acquisition Timing ... 58
Line Trigger Source Selection for Linescan Applications ... 59
CORACQ_PRM_EXT_LINE_TRIGGER_SOURCE – Parameter Values Specific to the X64-CL iPro series ... *59*
Shaft Encoder Interface Timing ... 61
Virtual Frame Trigger for Linescan Cameras ... 62
Acquisition Methods ... 64
X64-CL iPro LUT availability ... 65
X64-CL iPro\Lite LUT availability ... 66
Trigger-to-Image Reliability ... 67
Supported Events and Transfer Methods ... *67*
X64-CL iPro Sapera Parameters ... 69
Camera Related Parameters ... *70*
VIC Related Parameters ... *75*
ACQ Related Parameters ... *79*
X64-CL iPro Memory Error with Area Scan Frame Buffer Allocation ... 80

X64-CL IPRO SAPERA SERVERS & RESOURCES **81**
Servers and Resources ... 81
Transfer Resource Locations ... 82

TECHNICAL SPECIFICATIONS **83**
X64-CL iPro Board Specifications ... 83
X64-CL iPro\Lite Board Specifications ... 85
Host System Requirements ... 86
EMI Certifications ... 87
Connector and Switch Locations ... 88
X64-CL iPro Board Series Layout Drawings ... *88*
Connector Description List ... *89*
Connector and Switch Specifications ... 90
X64-CL iPro End Bracket ... *90*
X64-CL iPro\Lite End Bracket ... *90*
Status LEDs Functional Description ... *91*
J1: Camera Link Connector 1 (applies to X64-CL iPro and X64-CL iPro\Lite) ... *92*
J2: Camera Link Connector 2 (on X64-CL iPro with Dual Base Configuration) ... *92*
J2: Camera Link Connector 2 (on X64-CL iPro in Medium Configuration) ... *93*
Camera Link Camera Control Signal Overview ... *94*
J4: External Signals Connector ... *95*
X64-CL iPro: External Signals Connector Bracket Assembly ... *98*
Connecting a TTL Shaft Encoder Signal to the LVDS/RS422 Input ... *99*
External Trigger TTL Input Electrical Specification ... *101*

Strobe TTL Output Electrical Specification *101*
J7: Board Sync *102*
J8: Power to Camera Voltage Selector *102*
J9: PC Power to Camera Interface *103*
J11: Start Mode *103*
J3, J12: Reserved *103*
Brief Description of Standards RS-232, RS-422, & RS-644 (LVDS) *104*

CAMERA LINK INTERFACE **105**
CAMERA LINK OVERVIEW 105
Rights and Trademarks *105*
DATA PORT SUMMARY 106
CAMERA SIGNAL SUMMARY 106
CAMERA LINK CABLES 107

APPENDIX: X-I/O MODULE OPTION **109**
X-I/O MODULE OVERVIEW 109
X-I/O Module Connector List & Locations *110*
X-I/O MODULE INSTALLATION 110
Board Installation *111*
X64-CL iPro and X-I/O Driver Update *111*
X-I/O MODULE EXTERNAL CONNECTIONS TO THE DB37 111
DB37 Pinout Description *112*
Outputs in NPN Mode: Electrical Details *113*
Outputs in PNP Mode: Electrical Details *114*
Opto-coupled Input: Electrical Details *115*
TTL Input Electrical Details *115*
X-I/O MODULE SAPERA INTERFACE 116
Configuring User Defined Power-up I/O States *116*
Using Sapera LT General I/O Demo *117*
Sapera LT General I/O Demo Code Samples *120*

DALSA CONTACT INFORMATION **125**
SALES INFORMATION 125
TECHNICAL SUPPORT 126

GLOSSARY OF TERMS **127**

INDEX **129**

Introduction

Overview of the Manual

The X64-CL iPro Series *User's Manual* covers the following topics:

- **Overview of the X64-CL iPro Series**
  Description of the X64-CL iPro and X64-CL iPro\Lite products and a brief summary of their capabilities.
- **Installing the X64-CL iPro**
  Installation procedures for the X64-CL iPro board and driver under Windows XP, Windows Vista or Windows 7, as well as information on camera connectivity.
- **CamExpert Quick Start for the X64-CL iPro**
  User's guide to interfacing cameras with CamExpert and using the CamExpert demo tools.
- **The Sapera Demo Application**
  Using the Sapera Acquisition demo programs to test the **X64-CL iPro** installation.
- **X64-CL iPro Reference**
  Descriptions of **X64-CL iPro** and **X64-CL iPro\Lite** hardware, block diagram, capabilities, and acquisition modes supported.
- **X64-CL iPro Sapera Servers & Resources**
  Specifications specific to the Sapera Imaging Library.
- **Technical Specifications**
  Connector locations and pin-out descriptions.
- **Camera Link Interface**
  Overview of the Camera Link specification.
- **X-I/O Module Option**
  Describes the X-I/O module, its configuration, cabling, and usage.
- **DALSA Contact Information**
  Phone numbers, web site, and important email addresses.

About the Manual

This manual exists in printed, compiled HTML help, and Adobe® Acrobat (PDF) formats. The help and PDF formats make full use of hypertext cross-references and include links to the **DALSA** home page on the Internet, located at http://www.imaging.com/, accessed using any web browser.

Using the Manual

File names, directories, and Internet sites will be in bold text
(e.g., **image2.bmp**, **c:\sapera**, **http://www.imaging.com**).

Text that must be entered using the keyboard will be in typewriter-style text
(e.g., `c:\temp`).

Menu and dialog actions will be indicated in bold text in the order of the instructions to be executed, with each instruction separated by bullets. For example, going to the **File** menu and choosing **Save** would be written as **File•Save**.

The X64-CL iPro series consists of two product models, X64-CL iPro and X64-CL iPro\Lite. Unless a model is explicitly mentioned the name X64-CL iPro describes both models.

Overview of the X64-CL iPro Series

Product Part Numbers

X64-CL iPro Board Family

Item	Product Number
All models have 85MHz Pixel Clock X64-CL iPro with 32 MB of memory X64-CL iPro\Lite with 32 MB of memory	 OC-64E0-IPRO0 OC-64E0-IPROL
X-I/O Module (optional): provides 8 input & 8 output general I/Os (see "Appendix: X-I/O Module Option" on page 109)	OC-IO01-STD00
This manual in printed form, is available on request	OC-64EM-USER0

X64-CL iPro Software

Item	Product Number
Sapera LT version 5.20 or later (required but sold separately) 1. Sapera LT: Provides everything you will need to build your imaging application 2. Current Sapera compliant board hardware drivers 3. Board and Sapera documentation (compiled HTML help, and Adobe Acrobat® (PDF) formats)	OC-SL00-0000000
(optional) Sapera Processing Imaging Development Library includes over 600 optimized image processing routines.	Contact Sales at **DALSA**

X64-CL iPro Cables & Accessories

Item	Product Number
External Signals Connector Bracket Assembly supplied with each X64-CL iPro (connects to J4 – see "X64-CL iPro: External Signals Connector Bracket Assembly" on page 98) Note: not used when the X-I/O Module is installed.	OC-64CC-0TIO1
(optional) Power interface cable required when supplying power to cameras	OR-COMC-POW03
(optional) Camera Link Video Input Cable: 1 meter 2 meter	 OC-COMC-CLNK0 OC-COMC-CLNK6
(optional) DB25 male to color coded blunt end cable – 6 foot (1.82 meter) length	OC-COMC-XEND1

About the X64-CL iPro Series of Frame Grabbers

X64-CL iPro Series Key Features

Two Models are available:

- **X64-CL iPro\Lite**:
  Supports 1 Base Camera plus input lookup tables.
- **X64-CL iPro**:
  Supports Medium Monochrome Camera Link with Flat Field Correction
  Supports Medium RGB Camera Link
  Supports Base Camera Link with Bayer Filter Decoding
  Support 2 Base Monochrome Camera Link with Flat Field Correction
  Support 2 Base RGB Camera Link plus input lookup tables

See "Technical Specifications" on page 83 for detailed information.

X64-CL iPro User Programmable Configurations

Using the X64-CL iPro firmware loader utility, firmware for one of the supported modes is easily installed, either during driver installation or manually later on.

For the **X64-CL iPro** board the firmware choices are:

- **Two Base Camera Link Inputs with Flat Field Correction**
  Support for two independent Base Camera Link ports. Flat Field Correction (FFC) includes Fixed Pattern Noise (FPN), Pixel Replacement, Photo Response Non Uniformity (PRNU), and Shading Correction.
- **One Medium Camera Link Input with Flat Field Correction**
  Support for one Base or one Medium Camera Link port. Flat Field Correction (FFC) includes Fixed Pattern Noise (FPN), Pixel Replacement, Photo Response Non Uniformity (PRNU), and Shading Correction.
- **One Base Camera Link Input with Bayer Decoder**
  Support for one Base Camera Link port with Hardware Bayer CFA (Color Filter Array) Decoder. No Flat Field Correction is available.

Important: The **X64-CL iPro\Lite** product does not support alternative firmware choices with onboard processing.

ACUPlus: Acquisition Control Unit

ACUPlus consists of two sets of independent grab controllers, one pixel packer, and one time base generator. ACUPlus delivers a flexible acquisition front end plus it supports pixel clock rates of up to 85MHz.

ACUPlus acquires variable frame sizes up to 256KB per horizontal line and up to 16 million lines per frame. ACUPlus can also capture an infinite number of lines from a linescan camera without losing a single line of data.

ACUPlus supports standard Camera Link multi-tap configurations from 8 to 24-bit/pixels. Additionally, alternate tap configurations can support up to 4 taps of 8-bits each.

The X64-CL iPro model supports two cameras with different tap configurations simultaneously such as one dual tap 8/10/12-bits monochrome camera with an RGB 24-bit camera.

Camera Link Maximum Acquisition Rates:

This table specifies the X64-CL iPro sustainable acquisition hardware maximums, not the maximum data transfer rate through the PCI 64 bus to system memory.

Cameras connected	Camera Link standard	Maximum Acquisition rate 85 MHz components
2	Base	300 Mbytes/sec
1	Medium	300 Mbytes/sec

DTE: Intelligent Data Transfer Engine

The X64-CL iPro intelligent Data Transfer Engine ensures fast image data transfers between the board and the host computer with zero CPU usage. The DTE provides a high degree of data integrity during continuous image acquisition in a non-real time operating system like Windows. DTE consists of multiple independent DMA units, Tap Descriptor Tables, and Auto-loading Scatter-Gather tables.

Advanced Controls Overview

Visual Indicators

X64-CL iPro features a unique visual LED indicator to facilitate system installation and setup. This provides visual feedback indicating when the camera is connected properly and sending data.

External Event Synchronization

Two sets of dedicated trigger inputs and strobe signals are provided to synchronize precisely image captures with external events.

Camera Link Communications ports

Two PC independent communication ports provide Camera Link controls for camera configurations. These ports do not require additional PC resources like free interrupts or I/O address space. Accessible via the board device driver, the communication ports preset a seamless interface to Windows-based standard communication applications like HyperTerminal, etc. The communication ports are accessible directly from the Camera Link connectors.

Quadrature Shaft Encoder

Important feature for web scanning applications, the Quadrature-Shaft-Encoder inputs allow synchronized line captures from external web encoders.

PCI 64 bit Interface

The X64-CL iPro is a universal PCI slot board, compliant with the PCI version 2.2 specification for 64 bit devices. Transfer rates up to 528 Mbytes/sec with the appropriate camera are possible. The X64-CL iPro board occupies one computer expansion slot and one chassis opening.

About the X-I/O Module

The optional X-I/O module adds general purpose software controllable I/O signals to the X64-CL iPro. The X-I/O module provides 2 opto-coupled inputs, 6 logic signal inputs (5V or 24V), and 8 TTL outputs (NPN or PNP type selectable). All inputs or outputs provide a more electrically robust interface to outside devices offering better protection against real-world conditions. The module also makes available 5V or 12V dc power from the host system.

The X-I/O module can be either purchased with the X64-CL iPro board or installed into the computer system at a later time. The module occupies one adjacent PCI slot and connects to the X64-CL iPro via a ribbon cable. X-I/O Module external connections are made via the DB37 connector on the module bracket.

X-I/O requires X64-CL iPro board driver version 1.10 or later and Sapera LT version 5.30 or later. See "Appendix: X-I/O Module Option" on page 109 for details and specifications.

Development Software Overview

Sapera LT Library

Sapera LT is a powerful development library for image acquisition and control. Sapera LT provides a single API across all current and future DALSA hardware. Sapera LT delivers a comprehensive feature set including program portability, versatile camera controls, flexible display functionality and management, plus easy to use application development wizards.

Sapera LT comes bundled with CamExpert, an easy to use camera configuration utility to create new, or modify existing camera configuration files.

Sapera Processing Library

Sapera Processing is a comprehensive set of C++ classes for image processing and analysis. Sapera Processing offers highly optimized tools for image processing, blob analysis, search (pattern recognition), OCR and barcode decoding.

Installing the X64-CL iPro

Warning! (Grounding Instructions)

Static electricity can damage electronic components. Please discharge any static electrical charge by touching a grounded surface, such as the metal computer chassis, before performing any hardware installation.

If you do not feel comfortable performing the installation, please consult a qualified computer technician.

Important: Never remove or install any hardware component with the computer power on. Disconnect the power cord from the computer to disable the power standby mode. This prevents the case where some computers unexpectedly power up when a board is installed.

Upgrading Sapera or any DALSA Board Driver

When installing a new version of Sapera or a DALSA acquisition board driver in a computer with a previous installation, the current version **must** be un-installed first. Upgrade scenarios are described below.

Board Driver Upgrade Only

Minor upgrades to acquisition board drivers are typically distributed as ZIP files available in the DALSA web site http://www.imaging.com/downloads. Board driver revisions are also available on the next release of the Sapera CD-ROM.

Often minor board driver upgrades do not require a new revision of Sapera. To confirm that the current Sapera version will work with the new board driver:

- Check the new board driver ReadMe.txt file before installing, for information on the minimum Sapera version required.
- If the ReadMe.txt file does not specify the Sapera version, you can contact DALSA Technical Support (see "Technical Support" on page 126).

To upgrade the board driver only:

- Log on to the computer as an administrator or with an account that has administrator privileges.
- In **Windows XP**, from the start menu select **Start • Control Panel • Add or Remove Programs**. Select the DALSA X64-CL iPro board driver and click **Remove**.
- **Windows XP** only:
  - When the driver un-install is complete, reboot the computer.

- Log on to the computer as an administrator again.

- In **Windows Vista and Windows 7**, from the start menu select **Start • Control Panel • Programs and Features**. Double-click the DALSA X64-CL iPro board driver and click **Remove**.
- Install the new board driver. Run **Setup.exe** if installing manually from a downloaded driver file.
- If the new driver is on a Sapera CD-ROM follow the installation procedure described in "Installing X64-CL iPro Hardware and Driver" on page 11.
- Note that you can not install a DALSA board driver without Sapera LT installed on the computer.

Sapera and Board Driver Upgrades

When both Sapera and the DALSA acquisition board driver are upgraded, follow the procedure described below.

- Log on to the computer as an administrator or with an account that has administrator privileges.
- In **Windows XP**, from the start menu select **Start • Control Panel • Add or Remove Programs**. Select the DALSA X64-CL iPro board driver and click Remove. Follow by also removing the older version of Sapera LT.
- In **Windows Vista and Windows 7**, from the start menu select **Start • Control Panel • Programs and Features**. Double-click the DALSA X64-CL iPro board driver and click Remove. Follow by also removing the older version of Sapera LT.
- Reboot the computer and log on to the computer as an administrator again.
- Install the new versions of Sapera and the board driver as if this was a first time installation. See "Sapera LT Library Installation" on page 10 and "Installing X64-CL iPro Hardware and Driver" on page 11 for installation procedures.

Sapera LT Library Installation

Note: to install Sapera LT and the X64-CL iPro device driver, log on to the computer as an administrator or with an account that has administrator privileges.

The Sapera LT Development Library (or 'runtime library' if application development is not being performed) must be installed before the X64-CL iPro device driver.

- Insert the DALSA Sapera CD-ROM. If **AUTORUN** is enabled on your computer, the DALSA installation menu is presented.
- If **AUTORUN** is not enabled, use Windows Explorer and browse to the root directory of the CD-ROM. Execute **launch.exe** to start the DALSA installation menu and install the required Sapera components.
- The installation program will prompt you to reboot the computer.

Refer to *Sapera LT User's Manual* for additional details about Sapera LT.

Installing X64-CL iPro Hardware and Driver

In a Windows XP, Windows Vista or Windows 7 System

- Turn the computer off, disconnect the power cord (disables power standby mode), and open the computer chassis to allow access to the expansion slot area.
- Install the X64-CL iPro into a free 64 bit PCI expansion slot. If no 64 bit PCI slot is available, use a common 32 bit PCI slot. X64-CL iPro supports the plug and play automatic configuration of the PCI specification.
- Close the computer chassis and turn the computer on. Driver installation requires administrator rights for the current user of the computer.
- Windows will find the X64-CL iPro and start its **Found New Hardware Wizard**. Click on the **Cancel** button to close the Wizard Application.
- Insert the **DALSA Sapera CD-ROM**. If **AUTORUN** is enabled on your computer, the DALSA installation menu is presented. Install the X64-CL iPro driver.
- If **AUTORUN** is not enabled, use Windows Explorer and browse to the root directory of the CD-ROM. Execute **launch.exe** to start the DALSA installation menu and install the X64-CL iPro driver. Note, if you are using Windows Vista or Windows 7 with the User Account Control feature enabled, a dialog is displayed when you execute launch.exe; click **Allow** to continue with the driver installation.
- The driver automatically detects if a X64-CL iPro or X64-CL iPro\Lite board is installed. Choose the device driver setup type, full installation (required for application development) or runtime installation (supports application execution only).
- When using **Windows XP**, if a message stating that the X64-CL iPro software has not passed **Windows Logo testing** is displayed, click on **Continue Anyway** to finish the X64-CL iPro driver installation. Reboot the computer when prompted.
- When using **Windows Vista or Windows 7**, a message asking to install the DALSA device software is displayed. Click **Install**.

X64-CL iPro Device Driver Setup
Windows Security
Would you like to install this device software?
Name: X64-CL iPro Device Driver
Publisher: DALSA Corp.
Always trust software from "DALSA Corp.".
Install
Don't Install
You should only install driver software from publishers you trust. How can I decide which device software is safe to install?

- During the installation, the Device Manager firmware loader application starts. Click Update All. For more information on the Device Manager application, see "Displaying X64-CL iPro Board Information" on page 16.

X64-CL iPro Firmware Loader

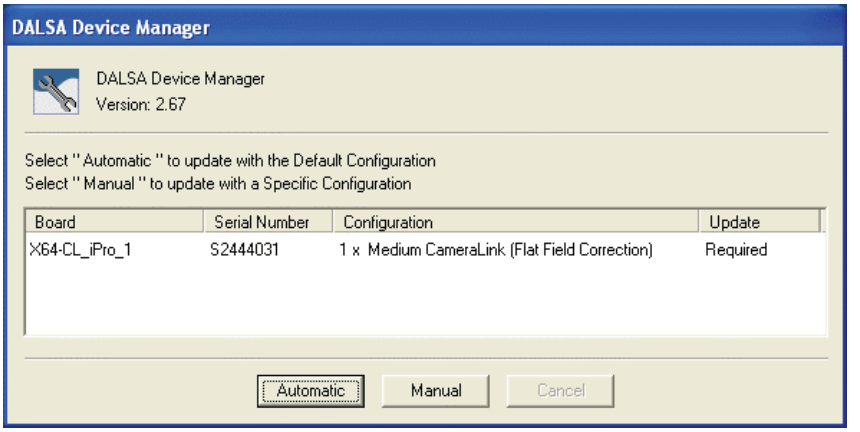
After Windows boots, the Device Manager program automatically executes. It determines if the installed driver is more recent than the current X64-CL iPro firmware. A dialog is presented to automatically update with default firmware or to manually select from optional firmware. Automatic and manual firmware updates are described in this section.

Important: In the vary rare case of firmware loader errors please see "Recovering from a Firmware Update Error" on page 21.

Automatic Mode

Click **Automatic** to update the X64-CL iPro firmware. The X64-CL iPro board supports three firmware configurations with the default being 2 x Base Camera Link cameras with Flat Field correction mode. See “X64-CL iPro Series Key Features” on page 5 and “X64-CL iPro User Programmable Configurations” on page 5 for details on all supported modes, which can be chosen via a manual update.

If there are multiple X64-CL iPro boards in the system, all boards will be updated with new firmware. If any installed X64-CL iPro boards installed in a system already have the correct firmware version, an update is not required. In the following screen shot, a single X64-CL iPro board is installed in the system and the default Medium configuration is ready to be programmed.

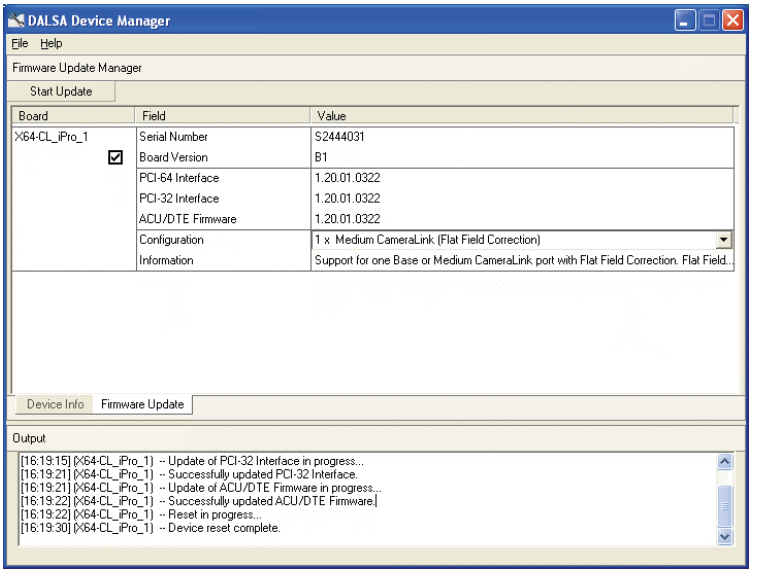


Manual Mode

Select **Manual** mode to load firmware other then the default version or when, in the case of multiple X64-CL iPro boards in the same system, each requires different firmware. The figure below shows the Device Manager manual firmware screen. Information on all installed X64-CL iPro boards, their serial numbers, and their firmware components are shown.

A manual firmware update is as follows:

- Select the X64-CL iPro via the board selection box (if there are multiple boards in the system).
- From the Configuration field drop menu select the firmware version required.
- Click on the Start Update button.
- Observe the firmware update progress in the message output window.
- Close the Device manager program when the X64-CL iPro board reset complete message is shown.



Executing the Firmware Loader from the Start Menu

If required, the X64-CL iPro Firmware Loader program is executed via the Windows Start Menu shortcut **Start • Programs • DALSA • X64-CL iPro Device Driver • Firmware Update**.

Enabling the Camera Link Serial Control Port

The Camera Link cabling specification includes a serial communication port for direct camera control by the frame grabber (see "J1: Camera Link Connector 1 (applies to X64-CL iPro and X64-CL iPro\Lite)" on page 92). The X64-CL iPro driver supports this serial communication port either directly or by mapping it to a host computer COM port. Any serial port communication program, such as Windows HyperTerminal, can connect to the camera in use and modify its function modes via its serial port controls. The X64-CL iPro serial port supports communication speeds up to 115 kbps.

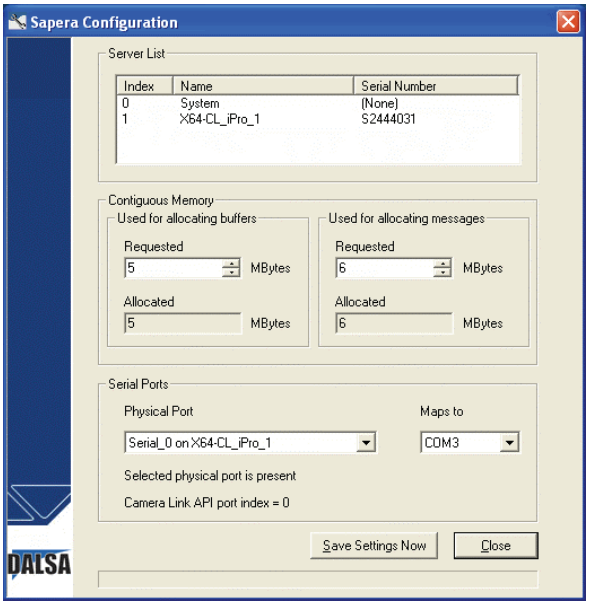
Note: if your serial communication program can directly select the X64-CL iPro serial port then mapping to a system COM port is not necessary.

The X64-CL iPro serial port is mapped to an available COM port by using the Sapera Configuration tool. Run the program from the Windows start menu: **Start • Programs • DALSA • Sapera LT • Sapera Configuration**.

COM Port Assignment

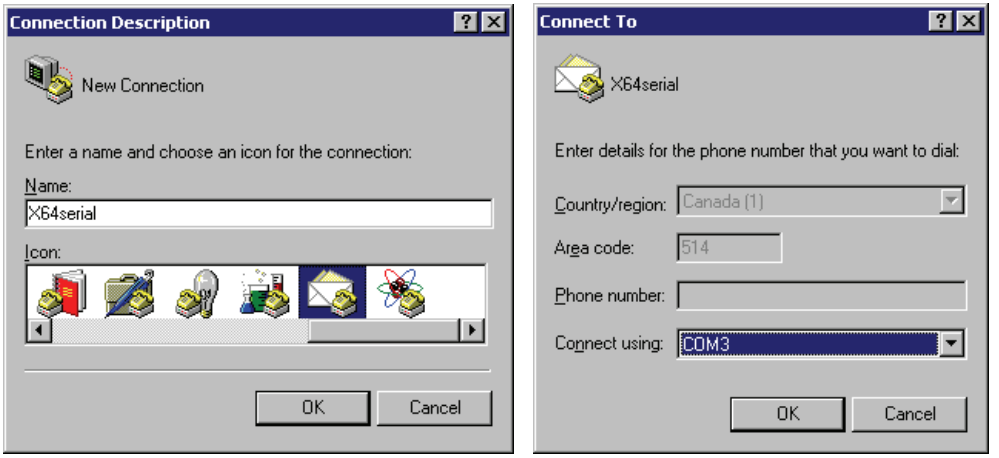
The lower section of the Sapera Configuration program screen contains the serial port configuration menu. Configure as follows:

- Use the **Physical Port** drop menu to select the Sapera board device from all available Sapera boards with serial ports (when more then one board is in the system).
- Use the **Maps to** drop menu to assign an available COM number to that Sapera board serial port.
- Click on the **Save Settings Now** button then the **Close** button. You are prompted to reboot your computer to enable the serial port mapping.
- The X64-CL iPro serial port, now mapped to COM3 in this example, is available as a serial port to any serial port application for camera control. Note that this serial port is not listed in the **Windows Control Panel•System Properties•Device Manager** because it is a logical serial port mapping.
- An example setup using Windows HyperTerminal follows (see "Setup Example with Windows HyperTerminal" on page 15).

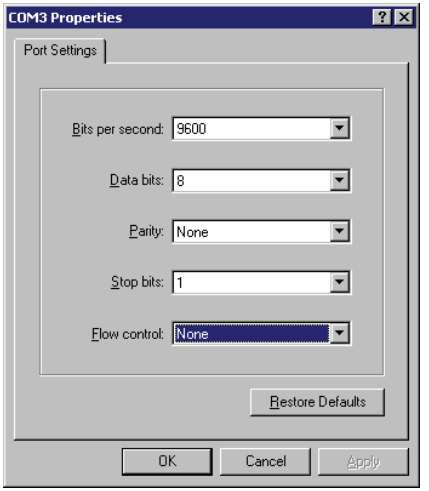


Setup Example with Windows HyperTerminal

- Run HyperTerminal and type a name for the new connection when prompted. Then click OK.
- On the following dialog screen select the COM port to connect with. In this example the X64-CL iPro serial port was previously mapped to COM3 by the Sapera Configuration program.



- HyperTerminal now presents a dialog to configure the COM port properties. Change settings as required by the camera you are connecting to. Note that the X64-CL iPro serial port does not support hardware flow control.



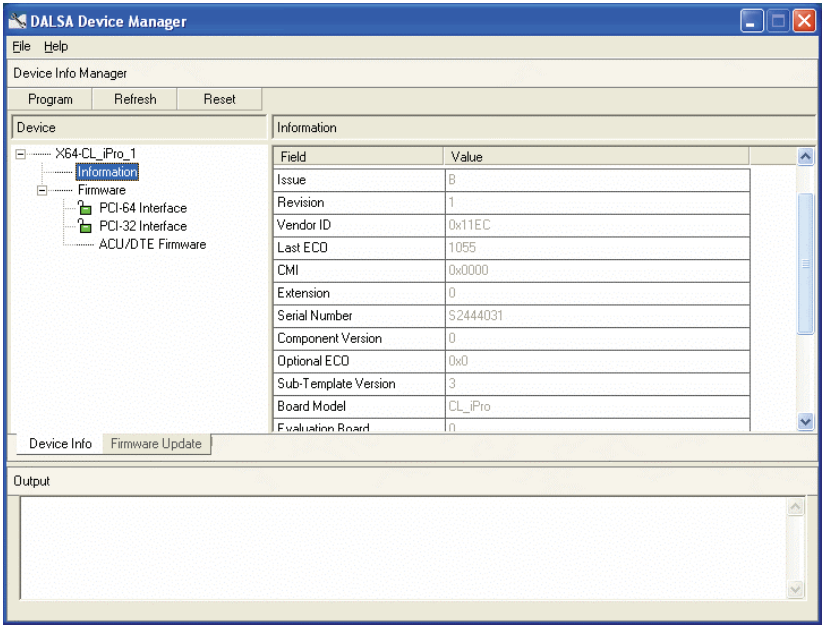
Displaying X64-CL iPro Board Information

The Device Manager program also displays information about the X64-CL iPro boards installed in the system. To view board information run the program via the Windows Start Menu shortcut **Start • Programs • DALSA • X64-CL iPro Device Driver • Device Manager**.

Device Manager – Board Information

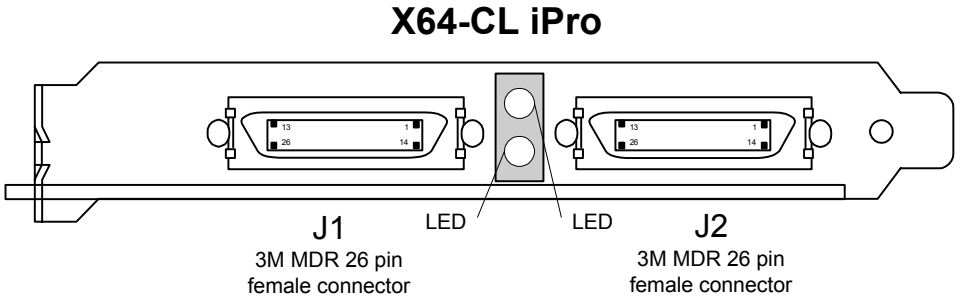
The following screen image shows the Device Manager program with Device Info tab active. The left pane displays all Sapera boards in the system. In this example there is only one X64-CL iPro. The right pane displays the information stored in the selected board device, which in this case is the information contained in the EEProm memory component.

A device manager report file (BoardInfo.txt) is generated by clicking **File • Save Device Info**. This report file may be requested by DALSA Technical Support to aid in troubleshooting installation or operational problems.



Camera to Camera Link Connections

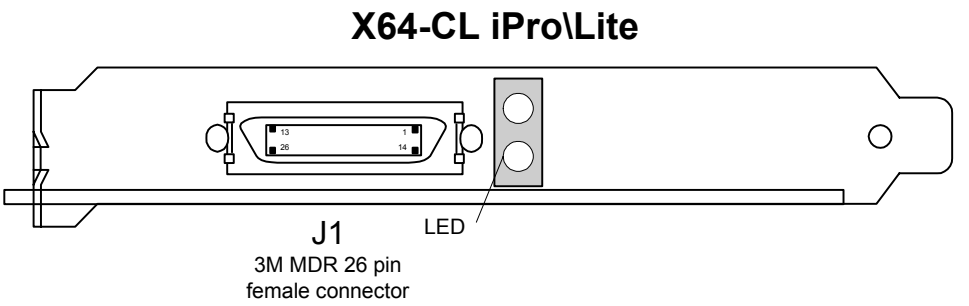
X64-CL iPro End Bracket



The hardware installation process is completed with the connection of a supported camera to the X64-CL iPro board using Camera Link cables (see "Camera Link Cables" on page **107**).

- The X64-CL iPro board supports a camera with one or two Camera Link MDR-26 connectors (two Base or one Medium – see "Data Port Summary" on page 106 for information on Camera Link configurations).
- Connect the camera to the J1 connector with a Camera Link cable. When using a Medium camera, connect the second camera connector to J2.

X64-CL iPro\Lite End Bracket



- The X64-CL iPro\Lite supports one Base Camera Link camera.
- Connect the camera to the J1 connector with a Camera Link cable.

Refer to section "Connector and Switch Specifications" on page **90** for details on the Camera Link connectors.

Caution: If the camera is powered by the X64-CL iPro, it is very important that the correct power supply voltage is selected. Refer to "J8: Power to Camera Voltage Selector" on page 102 for information on the voltage selection jumper.

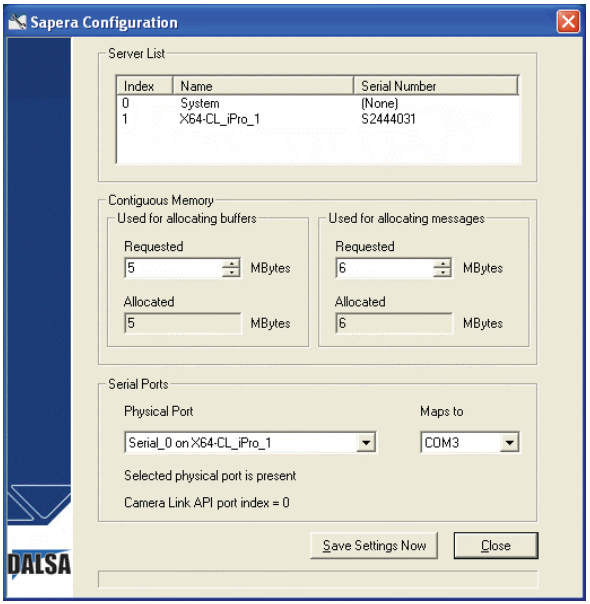
Contact DALSA or browse our web site http://www.imaging.com/camsearch for the latest information on X64-CL iPro supported cameras.

Configuring Sapera

Viewing Installed Sapera Servers

The Sapera configuration program (**Start • Programs • DALSA • Sapera LT • Sapera Configuration**) allows the user to see all available Sapera servers for the installed Sapera-compatible boards.

The **System** entry represents the system server. It corresponds to the host machine (your computer) and is the only server that should always be present. As shown in the following screen image, server index 1 is the installed X64-CL iPro board.



Increasing Contiguous Memory for Sapera Resources

The **Contiguous Memory** section lets the user specify the total amount of contiguous memory (a block of physical memory, occupying consecutive addresses) reserved for the resources needed for **Sapera buffers** allocation and **Sapera messaging**. For both items, the **Requested** value dialog box shows the driver default memory setting while the **Allocated** value displays the amount of contiguous memory that has been allocated successfully. The default values will generally satisfy the needs of most applications.

The **Sapera buffers** value determines the total amount of contiguous memory reserved at boot time for the allocation of dynamic resources used for host frame buffer management such as DMA descriptor tables plus other kernel needs. Adjust this value higher if your application generates any out-of-memory error while allocating host frame buffers. You can approximate the amount of contiguous memory required as follows:

- Calculate the total amount of host memory used for frame buffers
  (number of frame buffers • number of pixels per line • number of lines • (2 - if buffer is 10 or 12 bits)).
- Provide 1MB for every 256 MB of host frame buffer memory required.
- Add an additional 1 MB if the frame buffers have a short line length, say 1k or less
  (the increased number of individual frame buffers requires more resources).
- Add an additional 2 MB for various static and dynamic Sapera resources.
- Test for any memory error when allocating host buffers. Simply use the Buffer menu of the Sapera Grab demo program (see "Using the Grab Demo" on page 53) to allocate the number of host buffers required for your acquisition source. Feel free to test the maximum limit of host buffers possible on your host system – the Sapera Grab demo will not crash when the requested number of host frame buffers cannot be allocated.

Host Computer Frame Buffer Memory Limitations

When planning a Sapera application and its host frame buffers used, plus other Sapera memory resources, do not forget the Windows operating system memory needs. Window XP as an example, should always have a minimum of 128 MB for itself.

A Sapera application using *scatter gather buffers* could consume most of the remaining system memory. When using frame buffers allocated as a *single contiguous memory block*, typical limitations are one third of the total system memory with a maximum limit of approximately 100 MB. See the Buffer menu of the Sapera Grab demo program for information on selecting the type of host buffer memory allocation.

Contiguous Memory for Sapera Messaging

The current value for **Sapera messaging** determines the total amount of contiguous memory reserved at boot time for messages allocation. This memory space is used to store arguments when a Sapera function is called. Increase this value if you are using functions with large arguments, such as arrays and experience any memory errors.

Troubleshooting Installation Problems

The X64-CL iPro (and the X64 family of products) has been tested by DALSA in a wide variety of 64-bit and 32-bit PCI computers. Although unlikely, installation problems may occur due to the constant changing nature of computer equipment and operating systems. This section describes what the user can verify to determine the problem or the checks to make before contacting DALSA Technical Support. Note that information provided within this section will be updated with the latest information DALSA can provide for each manual version released.

If you require help and need to contact DALSA Technical Support, make detailed notes on your installation and/or test results for our technical support to review. See "Technical Support" on page 126 for contact information.

Recovering from a Firmware Update Error

This procedure is required if any failure occurred while updating the X64-CL iPro firmware on installation or during a manual firmware upgrade. On the rare occasion the board has corrupted firmware, any Sapera application such as CamExpert or the grab demo program will not find an installed board to control.

Possible reasons for firmware loading errors or corruption are:

- Computer system mains power failure or deep brown-out.
- PCI bus or checksum errors.
- PCI bus timeout conditions due to other devices.
- User forcing a partial firmware upload using an invalid firmware source file.

When the X64-CL iPro firmware is corrupted, executing a manual firmware upload will not work because the firmware loader can not communicate with the board. In an extreme case, corrupted firmware may even prevent Windows from booting.

***Solution*:** The user manually forces the board to initialize from write protected firmware designed only to allow driver firmware uploads. When the firmware upload is complete, the board is then rebooted to initialize in its normal operational mode.

- This procedure requires removing the X64-CL iPro board several times from the computer.
- *Important*: Referring to the board's user manual (in the connectors and jumpers reference section), identify the configuration jumper location. The Boot Recovery Mode jumper for the X64-CL iPro is J11 (see "J11: Start Mode" on page 103).
- Shut down Windows and power OFF the computer.
- Move the configuration switch for boot recovery (safe mode) from its default position to the boot recovery mode position.
- Power on the computer. Windows will boot normally.
- When Windows has started, do a manual firmware update procedure to update the firmware again (see "Executing the Firmware Loader from the Start Menu" on page 13).

- When the update is complete, shut down Windows and power off the computer.
- Set the Boot Recovery Mode switch back to its default position and reboot the computer once again.
- Verify that the frame grabber is functioning by running a Sapera application such as CamExpert. The Sapera application will now be able to communicate with the X64-CL iPro board.

Windows Event Viewer

Windows Event Viewer (**Computer Management • System Tools • Event Viewer**), lists various events that have taken place during the Operating System boot sequence. If a driver generates an error, it will normally log an entry in the event list.

Device Manager Program

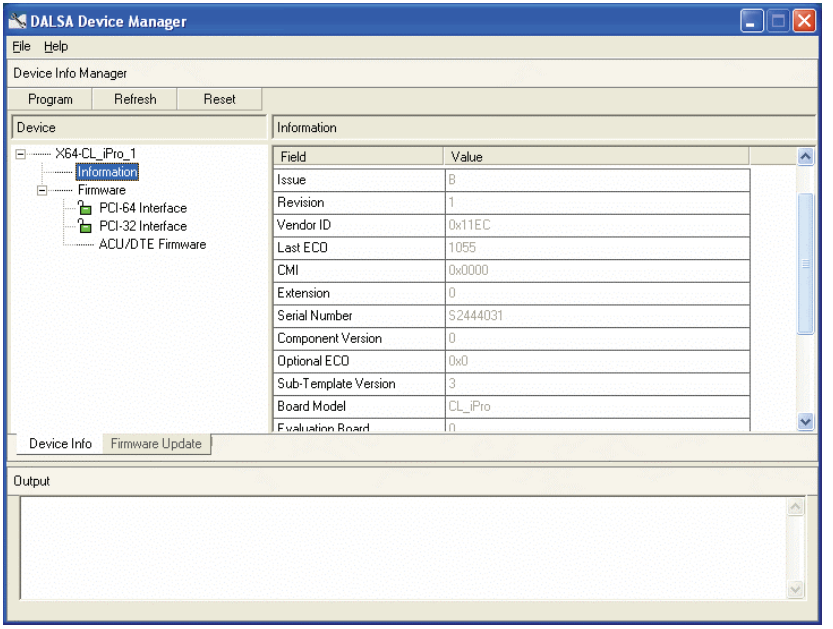
The Device Manager program provides a convenient method of collecting information about the installed X64-CL iPro. System information such as operating system, computer CPU, system memory, PCI configuration space, plus X64-CL iPro firmware information can be displayed or written to a text file (default file name – BoardInfo.txt). Note that this is a second function mode of the same program used to manually upload firmware to the X64-CL iPro.

Execute the program via the Windows Start Menu shortcut **Start • Programs • DALSA • X64-CL iPro Device Driver • Device Manager**. If the Device Manager program does not run, it will exit with a message that the board was not found. Since the X64-CL iPro board must have been in the system to install the board driver, possible reasons for an error are:

- Board was removed
- Board driver did not start or was terminated
- PCI conflict after some other device was installed

Information Window

The following figure shows the Device Manager information screen. Click to highlight one of the board components in the left pane and the information for that item is shown on the right.

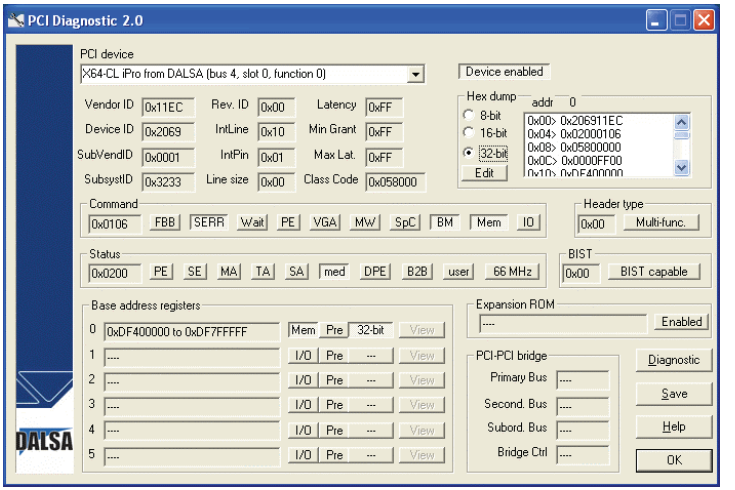


- Select **Information** to display identification and information stored in the X64-CL iPro firmware.
- Select one of the PCI interface components to load *custom* firmware when supplied by DALSA engineering for a future feature.
- Click on **File • Save Device Info** to save all information to a text file. Email this file when requested by Technical Support.

PCI Configuration

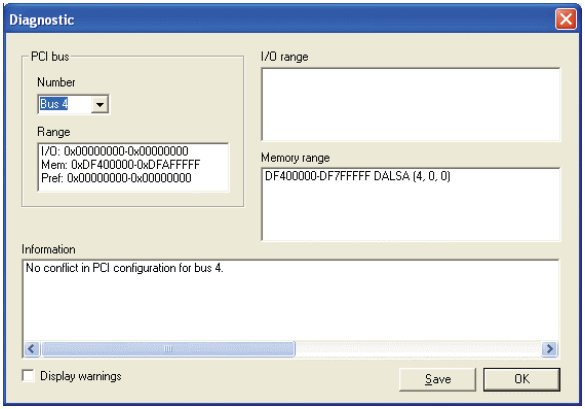
One of the first items to check when there is a problem with any PCI board is to examine the system PCI configuration and ensure that there are no conflicts with other PCI or system devices. The *DALSA PCI Diagnostic* program allows examination of the PCI configuration registers and can save this information to a text file. Run the program via the Windows Start Menu shortcut **Start • Programs • DALSA • Sapera LT • Tools • PCI Diagnostics**.

As shown in the following screen image, use the first drop menu to select the PCI device to examine. Select the device "X64-CL iPro from DALSA". Note the bus and slot number of the installed board (this will be unique for each system unless systems are setup identically). Click on the **Diagnostic** button to view an analysis of the system PCI configuration space.



Clicking on the **Diagnostic** button opens a new window with the diagnostic report. From the PCI Bus Number drop menu select the bus number that the X64-CL iPro is installed in. In this example the computer PCI expansion slots are identified as bus 4.

The window now shows the I/O and memory ranges used by each device on the selected PCI bus. The information display box will detail any PCI conflicts. If there is a problem, click on the **Save** button. A file named '**pcidiag.txt'** is created with a full dump of the PCI configuration registers. Email this file when requested by the DALSA Technical Support group along with a full description of your computer.

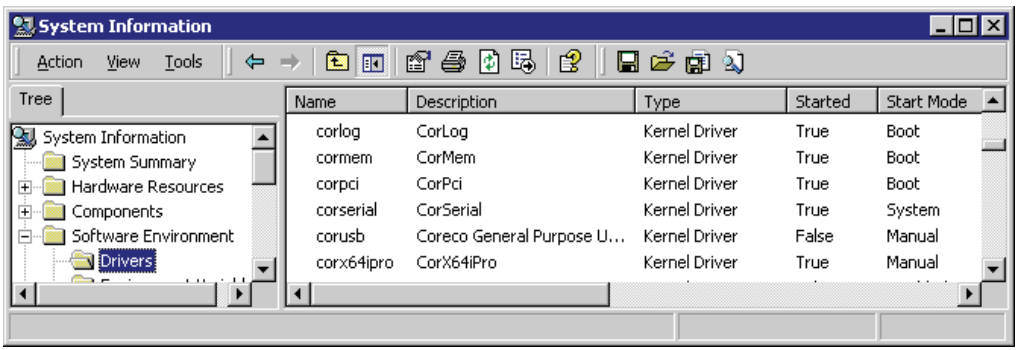


Sapera and Hardware Windows Drivers

The next step is to make certain the appropriate DALSA drivers have started successfully during the boot sequence. Example, click on the **Start • Programs • Accessories • System Tools • System Information • Software Environment**. Click on **System Drivers** (Windows XP). Make certain the following drivers have started for the X64-CL iPro.

Device	Description	Type	Started
CorX64iPro	X64-CL iPro driver	Kernel Driver	Yes
CorLog	Sapera Log viewer	Kernel Driver	Yes
CorMem	Sapera Memory manager	Kernel Driver	Yes
CorPci	Sapera PCI configuration	Kernel Driver	Yes
CorSerial	Sapera Serial Port manager	Kernel Driver	Yes

The **Drivers** dialog box should be similar to the following screenshot. All other drivers may differ on individual systems.



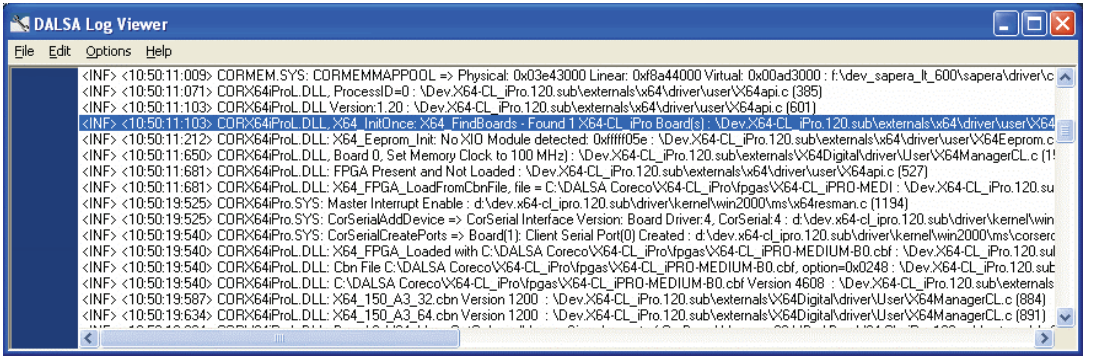
DALSA Technical Support may request that you check the status of these DALSA drivers as part of the troubleshooting process.

Log Viewer

The third step in the verification process is to save in a text file the information collected by the Log Viewer program. Run the program via the Windows Start Menu shortcut **Start • Programs • DALSA • Sapera LT • Tools • Log Viewer**.

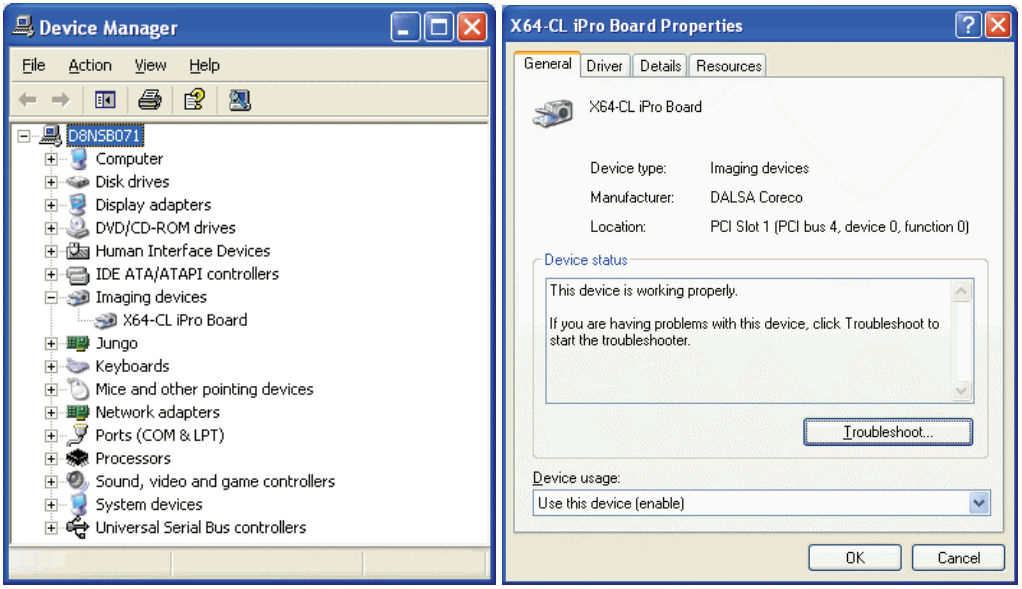
The Log Viewer lists information about the installed DALSA drivers. Click on **File • Save** and you will be prompted for a text file name to save the Log Viewer contents. Email this text file to DALSA Technical Support when requested or as part of your initial contact email.

Although the information collected by the Log Viewer seems complicated, you can make some initial diagnostics by checking the status of the DALSA driver. In the screen shot below, note the highlighted line which states [... CORX64iProL.DLL ... Found 1 X64-CL iPro board (s) ...]. This confirms that the driver can communicate with the X64-CL iPro.



Windows Device Manager

In Windows XP, use the Start Menu shortcut **Start • Settings • Control Panel • System • Hardware • Device Manager**. As shown in the following screen images, look for X64-CL iPro board under "Imaging Devices". Double-click and look at the device status. You should see "This device is working properly." Go to "Resources" tab and make certain that the device is mapped and has an interrupt assigned to it, without any conflicts.



Memory Requirements with Area Scan Acquisitions

The X64-CL iPro allocates by default two frame buffers in onboard memory, each equal in size to the acquisition frame buffer. This double buffering memory allocation is automatic at the driver level. The X64-CL iPro driver uses two buffers to ensure that the acquired video frame is complete and not corrupted in cases where the image transfer to host system memory may be interrupted and delayed by other host system processes. That is, the image acquisition to one frame buffer is not interrupted by any delays in transfer of the other frame buffer (which contains the previously acquired video frame) to system memory. Note that the number of onboard frame buffers is programmable.

The total size of the two internal frame buffers must be somewhat smaller than the total onboard memory due to memory overhead required for image transfer management. When the X64-CL iPro does not have enough onboard memory for the requested number of frame buffers, the memory error message [**Error: "CorXferConnect" <Xfer module> - No memory ()**] occurs when loading a Sapera camera file, or when the application configures a frame buffer.

Symptoms: CamExpert Detects no Boards

- If using Sapera version 5.20 or later:
  When starting CamExpert, if no DALSA board is detected, CamExpert will start in offline mode. There is no error message and CamExpert is functional for creating or modifying a camera configuration file. If CamExpert should have detected the installed board, troubleshoot the installation problem as described below.
- If using Sapera version 5.10:
  When starting CamExpert, you get this error window stating no DALSA product is found. Troubleshoot the installation problem as described below.

Troubleshooting Procedure

When CamExpert detects no installed DALSA board, there could be a hardware problem, a PnP problem, a PCI problem, a kernel driver problem, or a software installation problem.

- Make certain that the card is properly seated in PCI slot.
- Perform all installation checks described in this section ("Troubleshooting Installation Problems" on page 21) before contacting Technical Support.
- Try the board in a different PCI slot if it is not seen.

Symptoms: X64-CL iPro Does Not Grab

You are able to start Sapera CamExpert but you do not see an image and the frame rate displayed is 0.

- Verify power is connected to the camera.
- Verify the camera and timing parameters with the camera in free run mode.
- Make certain that you provide an external trigger if the camera configuration file requires one. Use the software trigger feature of CamExpert if you do not have a trigger source.
- Make certain that the camera is properly connected to the cable.
- Make certain that the camera is configured for the proper mode of operation. This must match the camera configuration file. Refer to your camera datasheet.
- Try to snap one frame instead of continuous grab.
- Perform all installation checks described in this section ("Troubleshooting Installation Problems" on page 21) before contacting Technical Support.

Symptoms: Card grabs black

You are able to use Sapera CamExpert, the displayed frame rate is as expected, but the display stays black.

- Set your camera to manual exposure mode and set the exposure to a longer period plus open the lens iris.
- Try to snap one frame instead of continuous grab.
- Make certain that the input LUT is not programmed to output all '0's.
- This problem is sometimes caused by a PCI transfer issue. No PCI transfer takes place, so the frame rate is above 0 but nevertheless no image is displayed in CamExpert.
- Make certain that BUS MASTER bit in the PCI configuration space is activated. Look in PCI Diagnostics for **BM** button under "Command" group. Make certain that the **BM** button is activated.
- Perform all installation checks described in this section ("Troubleshooting Installation Problems" on page 21) before contacting Technical Support.

Symptoms: Card acquisition bandwidth is less than expected

The X64-CL iPro is installed in a PCI-64 slot but the acquisition bandwidth is less than expected.

- Review the system for problems or conflicts with other expansion boards or drivers.
- Remove other PCI-32 or PCI-64 boards and check acquisition bandwidth again. Engineering has seen this case where other PCI boards in some systems cause limitations in PCI-64 transfers. Each system, with its combination of system motherboard and PCI boards, will be unique and will need to be tested for bandwidth limitations affecting the imaging application.

CamExpert Quick Start for the X64-CL iPro

Interfacing Cameras with CamExpert

CamExpert is the camera interfacing tool for frame grabber boards supported by the Sapera library. CamExpert generates the Sapera camera configuration file (*yourcamera.ccf*) based on timing and control parameters entered. For backward compatibility with previous versions of Sapera, CamExpert also reads and writes the *.cca and *.cvi camera parameter files.

An important component of CamExpert is its live acquisition display window which allows immediate verification of timing or control parameters without the need to run a separate acquisition program. Supplementing the live display is a status bar with information on video control signal. In the case of area scan digital cameras, the status bar informs the user about the presence of pixel clock, frame valid, and line valid signals.

Every Sapera demo program starts by a dialog window to select a camera configuration file. Even when using the X64-CL iPro with common video signals, a camera file is required. Therefore CamExpert is typically the first Sapera application run after an installation.

For context sensitive help click on the button then click on a camera configuration parameter. A short description of the configuration parameter will be shown in a popup. Click on the button to open the help file for more descriptive information on CamExpert.

CamExpert Demonstration and Test Tools

The CamExpert utility also includes a number of demonstration features which make CamExpert the primary tool to configure, test and calibrate your camera and imaging setup. Display tools include, image pixel value readout, image zoom, and line profiler.

Functional tools include hardware Flat Field calibration and operation support (see “Using the Flat Field Correction Tool” on page 48), plus support for either hardware based or software Bayer filter camera decoding with auto white balance calibration (see “Using the Bayer Filter Tool” on page 50).

Camera Types & Files Applicable to the X64-CL iPro

The X64-CL iPro supports digital area scan or linescan cameras using the Camera Link interface standard. See "Camera to Camera Link Connections" on page 17 for information on connecting a Camera Link camera.

Contact DALSA or browse our web site [http://www.imaging.com/camsearch] for the latest information and application notes on X64-CL iPro supported cameras.

Camera Files Distributed with Sapera

The Sapera distribution CDROM includes camera files for a selection of X64-CL iPro supported cameras. Using the Sapera CamExpert program, you may use the camera files (CCA) provided to generate a camera configuration file (CCF) that describes the desired camera and frame grabber configuration.

DALSA continually updates a camera application library composed of application information and prepared camera files. Along with the camera search utility on the DALSA web site, as described above, a number of camera files are ready to download from the Sapera Camera File FTP site [ftp://ftp.coreco.com/public/Sapera/CamFile_Updates]. Camera files are ASCII text and can be read with Windows Notepad on any computer without having Sapera installed.

CamExpert Memory Errors when Loading Camera Configuration Files

See section "Memory Requirements with Area Scan Acquisitions" on page 27.

Overview of Sapera Acquisition Parameter Files (*.ccf or *.cca/*.cvi)

Concepts and Differences between the Parameter Files

There are two components to the legacy Sapera acquisition parameter file set: CCA files (also called cam-files) and CVI files (also called VIC files, i.e. video input conditioning). The files store video-signal parameters (CCA) and video conditioning parameters (CVI), which in turn simplifies programming the frame-grabber acquisition hardware for the camera in use. **Sapera LT 5.0** introduces a new camera configuration file (**CCF**) that combines the CCA and CVI files into one file.

Typically, a camera application will use a CCF file per camera operating mode (or one CCA file in conjunction with several CVI files, where each CVI file defines a specific camera operating mode). An application can also have multiple CCA/CCF files so as to support different image format modes supported by the camera or sensor (such as image binning or variable ROI).

CCF File Details

Files using the ".CCF" extension, (Camera Configuration files), are essentially the camera (CCA) and frame grabber (CVI) parameters grouped into one file for easier configuration file management. This is the default Camera Configuration file used with Sapera LT 5.0 and the CamExpert utility.

CCA File Details

DALSA distributes camera files using the ".CCA" extension, (CAMERA files), which contain all parameters describing the camera video signal characteristics and operation modes (what the camera outputs). The Sapera parameter groups within the file are:

- Video format and pixel definition.
- Video resolution (pixel rate, pixels per line, lines per frame).
- Synchronization source and timing.
- Channels/Taps configuration.
- Supported camera modes and related parameters.
- External signal assignment.

CVI File Details

Legacy files using the ".CVI" extension, (Configuration Video files), contain all operating parameters related to the frame grabber board - what the frame grabber can actually do with camera controls or incoming video. The Sapera parameter groups within the file are:

- Activate and set any supported camera control mode or control variable.
- Define the integration mode and duration.
- Define the strobe output control.
- Allocate the frame grabber transfer ROI, the host video buffer size and buffer type (RGB888, RGB101010, MONO8, MONO16).
- Configuration of line/frame trigger parameters such as source (internal via the frame grabber /external via some outside event), electrical format (TTL, LVDS, OPTO-isolated), and signal active edge or level characterization.

Camera Interfacing Check List

Before undertaking the task of interfacing a camera from scratch with CamExpert:

- Confirm that DALSA has not already published an application note with camera files [http://www.imaging.com/camsearch].
- Confirm that the correct version or board revision of X64-CL iPro is used. Confirm that the required firmware is loaded into the X64-CL iPro.
- Confirm that Sapera does not already have a .cca file for your camera installed on your hard disk. If there is a .cca file supplied with Sapera, then use CamExpert to automatically generate the .ccf file with default parameter values matching the frame grabber capabilities.

- Check if the Sapera installation has a similar type of camera file. A similar .cca file can be loaded into CamExpert where it is modified to match timing and operating parameters for your camera, and lastly save them as Camera Configuration file (.ccf), or as a new .cca & .cvi camera file pair for applications built with Sapera 4.2 or earlier.
- Finally, if your camera type has never been interfaced, run CamExpert after installing Sapera and the acquisition board driver, select the board acquisition server, and enter the camera parameters.

Linescan Example: Interfacing the Dalsa Piranha2 Linescan Camera

These examples use a X64-CL iPro (medium configuration) board connected to the Piranha2 Camera Link camera (P2-2x or P2-4x). The model P2-4x 06k, a 4 tap 40 MHz 6k pixel digital linescan camera was used for the CamExpert screen shots. Download the user's manual directly from the Dalsa web site http://www.dalsa.com/.

CamExpert Interfacing Outline

This outline is an overview of the steps required to connect and test a camera. These steps are described in greater detail in the sections that follow.

- Sapera and the X64-CL iPro device driver are installed as described in this manual.
- Check for an existing CCA file, distributed with Sapera, that will provide camera timing parameters.
- Check the DALSA web site for a published application note with camera files [http://www.imaging.com/camsearch].
- Connect two (for this example) Camera Link interface cables between the X64-CL iPro and camera. Connect power to the camera.
- Run Windows HyperTerminal and establish communication with the camera. Verify the camera's settings (the Dalsa Piranha2 command to return all camera parameters is get_camera_parameters).
- Set the camera to its free run mode or its self generated test pattern. This will simplify testing the camera file timing parameters.
- Run CamExpert, load or set camera timing parameters and capabilities, and then test a live grab. Save the camera file for the default free run mode.
- Using HyperTerminal, set the camera to external sync and external line integration mode.
- Configure CamExpert for camera external sync with external line integration control. Test variations in line rate and integration period. Save a new camera file when satisfied.

Step 1: Piranha2 in Free Run Exposure Mode

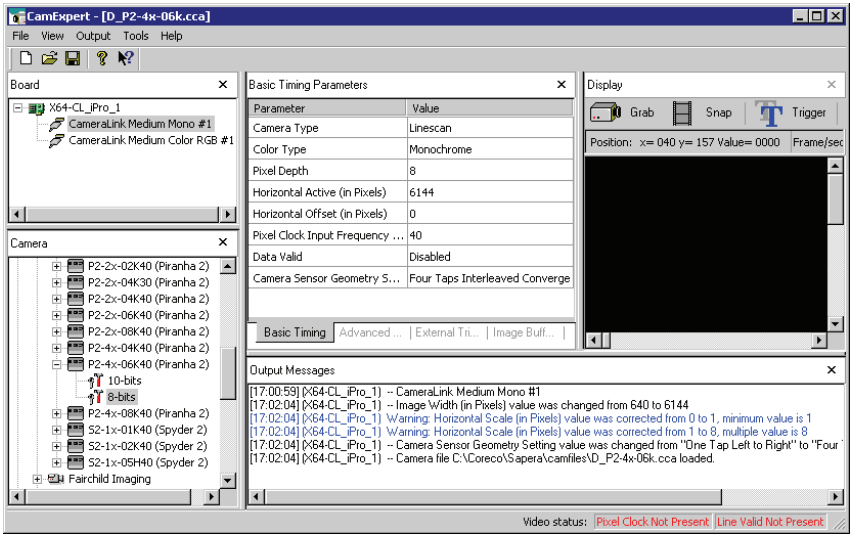
This section illustrates the CamExpert dialog screens for interfacing the Dalsa Piranha2 in free run mode. Dalsa defines this exposure mode as where the camera uses its internal SYNC and PRIN, at a maximum line rate and exposure time. This is the Piranha2 factory default mode and serves well as a CamExpert interfacing example.

To verify or set the Piranha2 in free run mode:

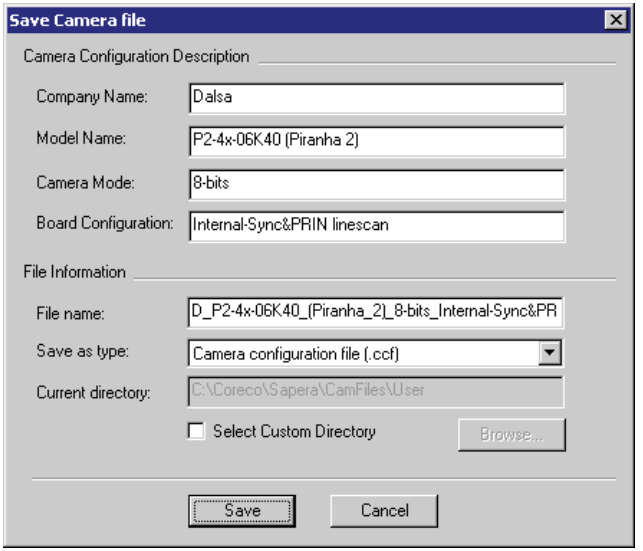
- With HyperTerminal type the Dalsa command to return the camera parameters get_camera_parameters.
- Check the value for Exposure Mode, which is 2 for internal SYNC and PRIN.
- If the value is different, enter the command set_exposure_mode 2.

File Selection & Grab Test

- From the Windows start menu run the Sapera CamExpert program. [**Programs|DALSA|Sapera LT|CamExpert**]
- CamExpert opens with default settings for the X64-CL iPro (assuming it is the only Sapera frame grabber installed). In the **Board** window, click on the Camera Link Medium Mono #1 acquisition server.
- The **Camera** window shows camera files distributed with Sapera that are supported by the X64-CL iPro (Camera Library). The User's Configuration File section is the default location for *.ccf files saved by CamExpert and also the default location that Sapera demo programs use to read camera configuration files.
- Select the Dalsa P2-4x 06k Piranha2 8-bit camera. The following screen image shows that CamExpert automatically loads the basic timing parameters from the distribution cca file.



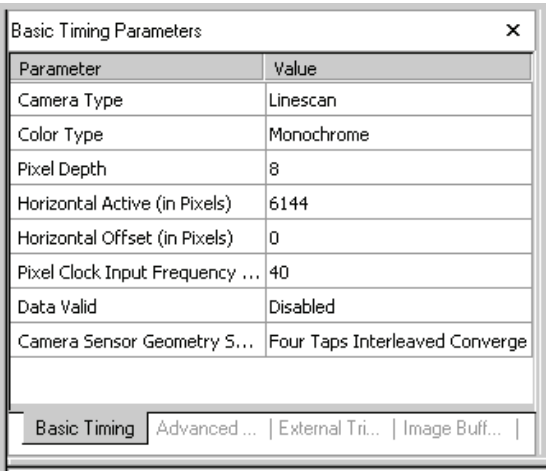
- With the Piranha2 configured for internal sync and PRIN (free run mode), click the Grab button to have live acquisition in the display window. The camera may need to be pointed to a bright wall or light source.
- Refer to the CamExpert video **status bar** to see if any required timing signals are missing. Also check the X64-CL iPro status LED (see "Status LEDs Functional Description" on page **91**) to aid in troubleshooting camera problems.
- This non-triggered exposure mode can be used to confirm a linescan inspection setup if the speed of the object or web is varied to match the fixed camera exposure.
- Use the **File-Save** dialog to save this camera configuration file (*.ccf) with user entered information. CamExpert provides information for each field based on the file originally loaded. Modify the fields such as camera mode and board configuration, to describe the parameter setup. Modify the file name and click Save. The ccf file is located in the default Sapera user folder. The screen image below is an example.



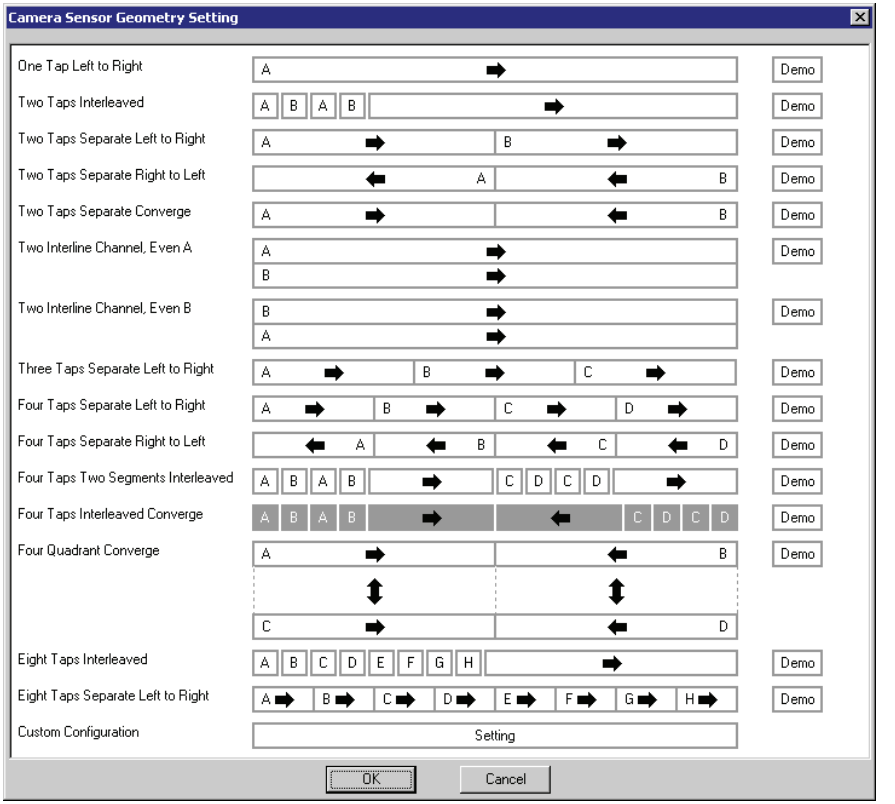
Overview of Basic Timing Parameters

CamExpert only shows parameters applicable to the acquisition board and camera type. When configuring parameters for a new camera start by selecting or entering the basic horizontal timing parameters and pixel clock frequency as defined by the camera manufacturer.

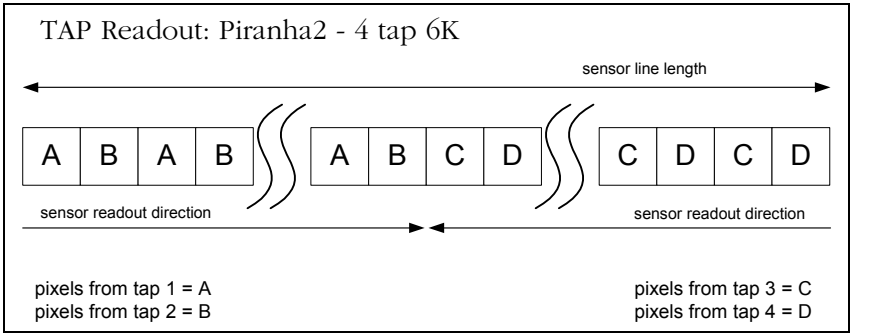
The following screen image shows the Basic Timing Parameters required for the X64-CL iPro with a linescan camera (parameters are for the Piranha2 - P2-4x 06k 8-bit camera).



- Pixel Depth depends on the camera digital data. The Dalsa Piranha2 linescan camera digitizes internally to 10 bits and outputs either all 10 bits or the most significant 8 bits. When selecting 8 bits, the Sapera frame buffer required is 8 bit mono. When selecting 10 bit, the Sapera frame buffer required is 16 bit mono.
- Data Valid: Some Camera Link cameras use a data valid signal in addition to line valid and/or frame valid control signals. The Dalsa Piranha2 does not, therefore this selection is set to disable.
- For the **Camera Sensor Geometry** parameter select one from the many standards supported by the acquisition board, or the user defines a custom geometry. The following screen image shows the sensor geometry selection window for the X64-CL iPro.


Camera Sensor Geometry Setting
One Tap Left to Right
Two Taps Interleaved
Two Taps Separate Left to Right
Two Taps Separate Right to Left
Two Taps Separate Converge
Two Interline Channel, Even A
Two Interline Channel, Even B
Three Taps Separate Left to Right
Four Taps Separate Left to Right
Four Taps Separate Right to Left
Four Taps Two Segments Interleaved
Four Taps Interleaved Converge
Four Quadrant Converge
Eight Taps Interleaved
Eight Taps Separate Left to Right
Custom Configuration
Setting
Demo
OK
Cancel

The Piranha2 model P2-4x 06k, 4 tap - 6k pixel camera uses a tap structure as shown in the next figure. Tap 1 and 2 are simultaneously readout from a left to right direction while taps 3 and 4 are readout from a right to left direction. The advantage of multiple tap cameras is that the pixel clock rate is kept reasonably low while the data output to the frame grabber is increased. A camera with more than one tap has a higher average bandwidth by simultaneously outputting separate portions of a single sensor exposure.


TAP Readout: Piranha2 - 4 tap 6K
sensor line length
A B A B
A B C D
C D C D
sensor readout direction
sensor readout direction
pixels from tap 1 = A
pixels from tap 2 = B
pixels from tap 3 = C
pixels from tap 4 = D

Step 2: Piranha2 in External Exposure Mode

Using the Piranha2 camera in external exposure (line integration) mode requires a few changes to both the camera's operating mode and the Piranha2 camera file used for free run exposure mode. The modified camera file should be given an appropriate description and saved with a unique file name.

Before using CamExpert to generate and test the modified camera file, set the Piranha2 camera to the desired external exposure mode as follows:

- With HyperTerminal type the Dalsa command set_exposure_mode 5 to set the camera for external SYNC and PRIN control from the frame grabber. Acquisition and exposure will now be controlled by the X64-CL iPro .
- The Piranha2 camera will respond with warnings that external SYNC and PRIN are not detected. This is normal since CamExpert has not yet been configured for line integration mode.

To simplify this example the X64-CL iPro will generate the line sync trigger. The X64-CL iPro is programmed for the line trigger frequency and line integration method matching the Piranha2 specifications.

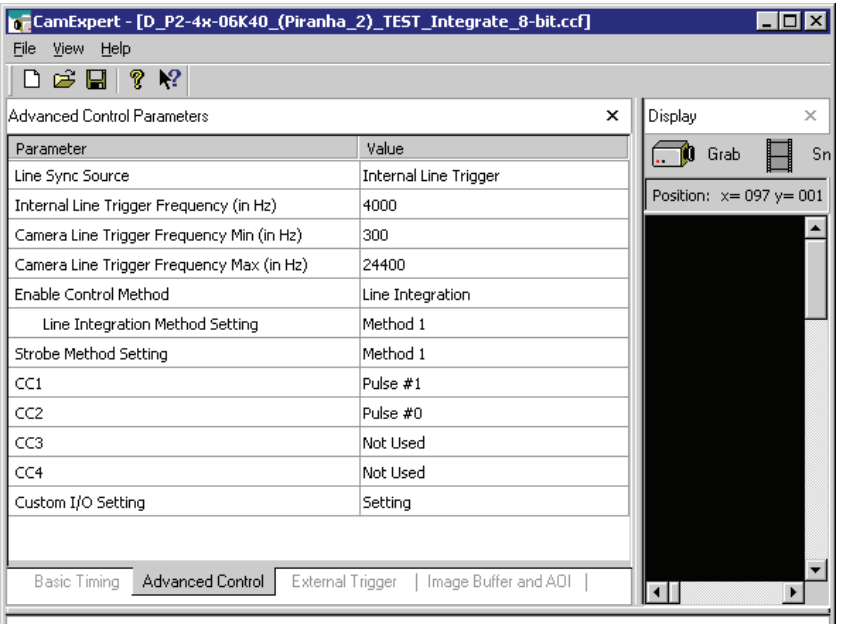
CCF File Selection

To configure this new CCF file, start by loading the CCF for free run mode from the previous section.

- From the Windows start menu run the Sapera CamExpert program.
  [**Programs|DALSA|Sapera LT|CamExpert**]
- In the **Board** window, click on the Camera Link Full Mono #1 acquisition server.
- From the **Camera** file selection window, select the ccf camera file previously saved for the Piranha2 in free run exposure mode, as configured in the previous section "Step 1: Piranha2 in Free Run Exposure Mode" on page 35.

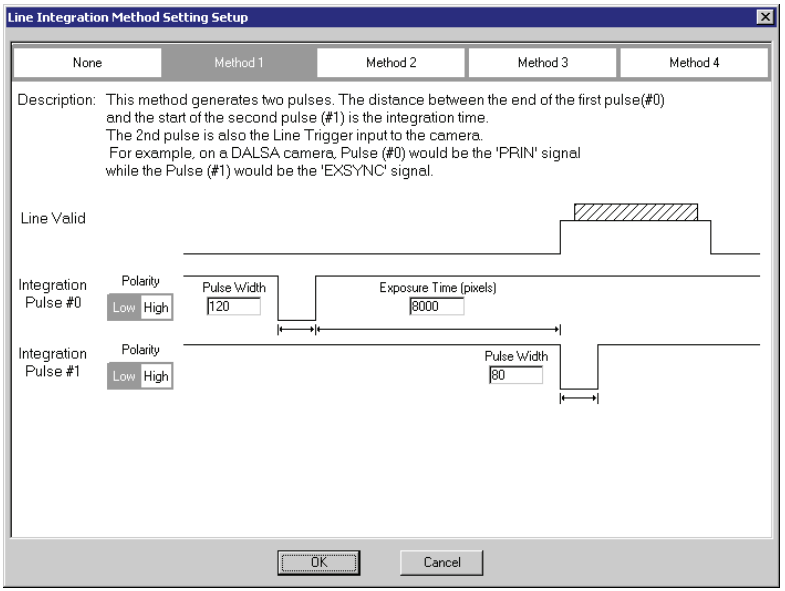
Advanced Control Parameters

The CamExpert advanced control parameters tab, as shown in the following screen image, contains the configuration items needed for external trigger and exposure control. Descriptions of each parameter, as setup for this example with the Piranha2, follow the screen image.



- **Line Sync Source**: Set to internal line trigger. The X64-CL iPro will generate the camera line trigger without using an external event or trigger.
- **Internal Line Trigger Frequency**: Set the line rate generated by the X64-CL iPro . The line sync rate is set to 4 kHz which is a period of 250µs. This period must be longer then the combined PRIN control pulse time (active low for pixel reset) and the integration time (time between rise of PRIN to EXSYNC). These control signals are described below.
- **Camera Line Trigger Frequency Min & Max**: Set to the camera specification limits so that CamExpert can trap invalid entries.
- **Enable Control Method**: Line Integration is selected as the required control mode.
- **Line Integration Method Setting and Camera Link Control Signals CC1-CC4**: The Camera Link CC1 and CC2 controls are assigned to Sapera controls which are not very descriptive by themselves. These selections are dependent on the camera's control specifications. The required camera controls are matched to the corresponding Sapera exposure method. The logical sequence to set these parameters is as follows.
  - From the Piranha2 user manual, the required Camera Link camera control configuration is **CC1 = EXSYNC** and **CC2 = PRIN**.

- From the Piranha2 user manual, note the timing requirements for external EXSYNC and PRIN. PRIN specifies a minimum logic low time for pixel reset which must be respected. When PRIN is logic high, the camera is integrating. Following an integration period the falling edge of EXSYNC triggers the line readout.
- Review the Sapera integration methods for a two pulse control which matches the camera control specifications. The Sapera Line Integration Method #1 matches the control requirement.
- From this information, it is seen that PRIN (CC2) corresponds to the Sapera control Pulse #0 and EXSYNC (CC1) corresponds to Sapera control Pulse # 1. Set the CamExpert camera controls to match these requirements. Note that Camera Link controls CC3 and CC4 are not used by the Piranha2.
- Click on the line integration method setting field. A configuration window allows selecting a method number corresponding to the Sapera integration methods.
- Select method 1. Configure the control pulses for polarity and signal width as shown in the following screen image. Descriptions for these parameters follow.



- For Pulse #0 (assigned to PRIN), select polarity and width. The Piranha2 specifies that PRIN must be a minimum 2µs active low pulse. At a 40Mhz pixel clock, 40 pixels equal 1µs, thus a setting of 120 pixels generates a 3µs PRIN. Note that the X64-CL iPro on-board pulse generator works in increments of 1µs.
- Knowing that 40 pixels is 1µs, this example sets an integration time of 8000 pixels which is 200µs. This value must be less then the line trigger rate which was set to 4 KHz (250µs).
- For Pulse #1 (assigned to EXSYNC), select polarity and width. In this example EXSYNC is active low for 2 µs (80 pixels).

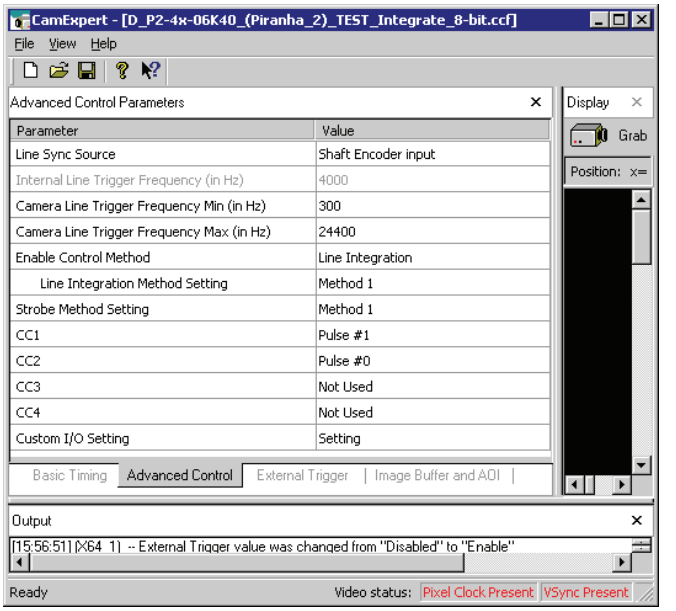
Step 3: Piranha2 with Shaft Encoder Line Sync

Continuing from the previous setup ("Step 2: Piranha2 in External Exposure Mode" on page 39), this section details using the X64-CL iPro shaft encode inputs as the exposure trigger for the imaging system. In addition the virtual frame reset feature is used to have an n number of image lines grabbed into the Sapera frame buffer when triggered by some external event.

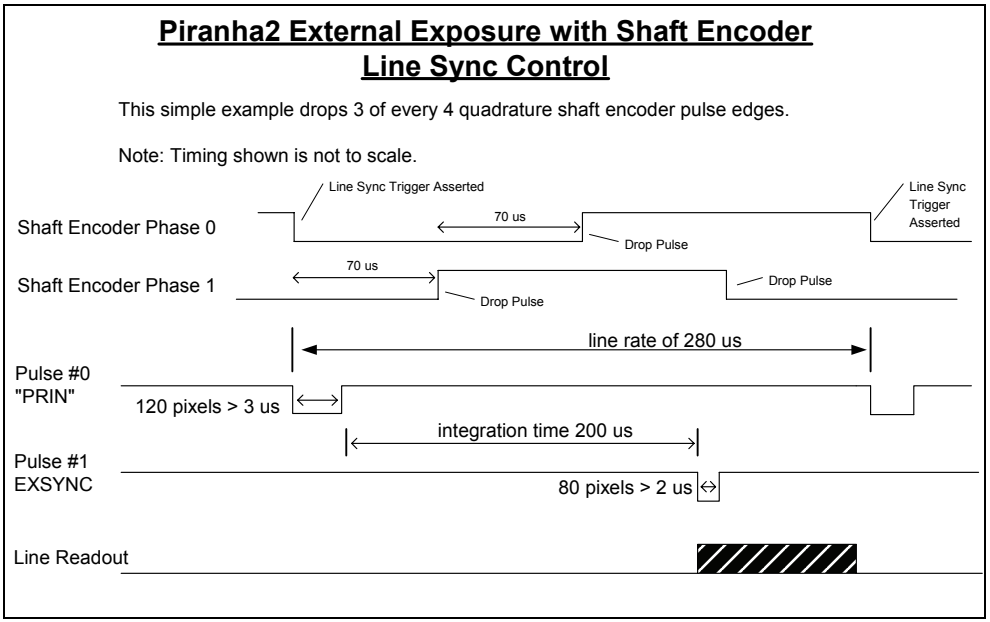
- See "Shaft Encoder Interface Timing" on page 61 for an overview of the quadrature shaft encoder supported by the X64-CL iPro , and the external signal connections used.
- See "Virtual Frame Trigger for Linescan Cameras" on page 62 for an overview of using an external frame reset signal to initiate the acquisition of n number of lines, and the external signal connections used.

Shaft Encoder Line Sync Setup

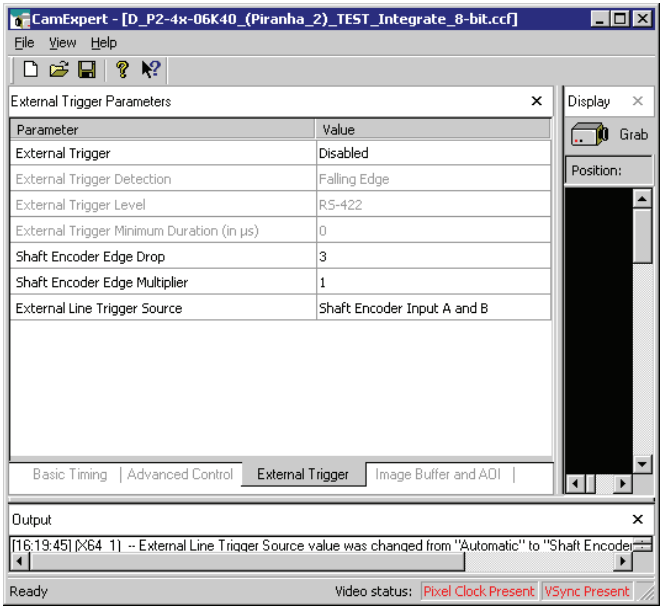
- Assuming the same PRIN and EXSYNC timing from the previous example, on the Advanced Control Parameters tab, the **Line Sync Source** signal Shaft Encoder is now selected.



- This example uses a line integration time of 200µs therefore the line sync source must have a period greater then the integration time plus the PRIN duration.
- Assuming quadrature shaft encoder signals every 70µs, by dropping three out of every four pulses, a line sync occurs every 280µs. The following timing diagram (not to scale) illustrates the shaft encoder signals relative to PRIN and EXSYNC.



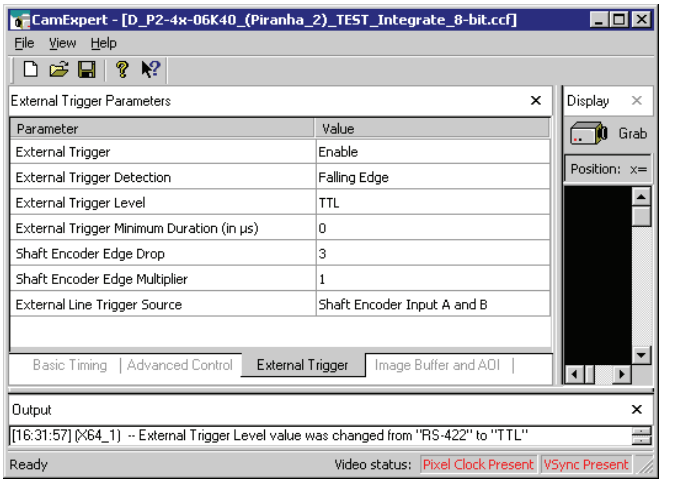
- Select the CamExpert **External Trigger Parameters** tab. The External Line Trigger Source is set to use both shaft encoder inputs. The edge drop factor is set to 3 to match the example timing diagram shown above.



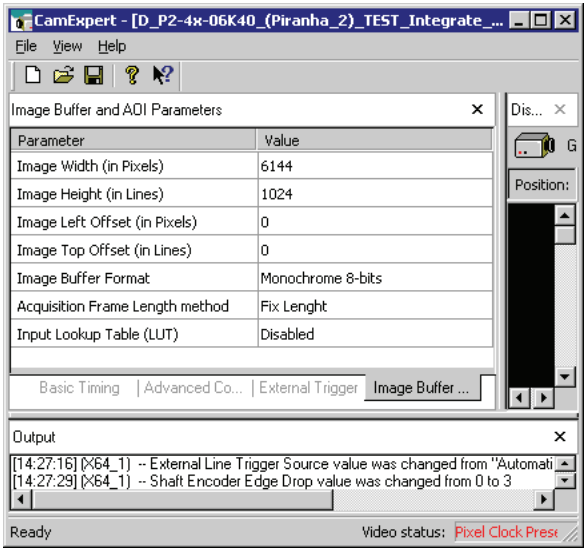
Using shaft encoder signals only, the X64-CL iPro grabs linescan data continuously into a memory frame buffer. The number of data lines stored (i.e. the vertical size of this frame buffer – see the Image Height parameter on the Image Buffer tab) is arbitrarily set by the user. With continuous shaft encode signals, after the frame buffer is filled with captured data lines, new data lines will then overwrite previous data.

Shaft Encoder with Fixed Frame Buffer Setup

To synchronize the capture of linescan data, an external trigger signal input (Min. pulse—100ns) to the X64-CL iPro is used to start acquisition into the frame buffer. In this example, when the external trigger is asserted, the X64-CL iPro then triggers linescan data captures based on the shaft encoder inputs. When the frame buffer is filled, linescan capture is suspended until the next external trigger. Example screen images of the External Trigger Parameters and the Image Buffer Parameters tabs follow.



- **External Trigger**: Set to Enable
- **External Trigger Detection**: Select which pulse edge or from a level active trigger. Available choices are dependent on the acquisition board used.
- **External Trigger Level**: Choose the signal type connect as an external trigger.



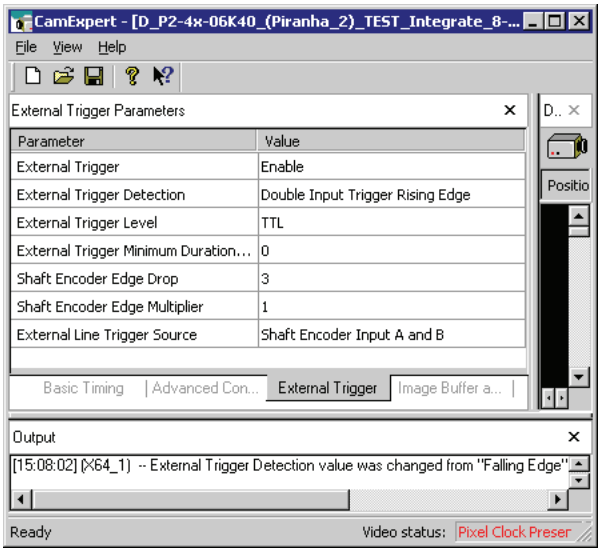
- **Image Width**: By default the buffer is the same as the acquisition width.
- **Image Height**: Set to the maximum number of acquired lines needed. Refer to the hardware specifications for the maximum limit.
- **Acquisition Frame Length method**: In this example, set to fixed length when there is a single trigger signal to start the acquisition of one complete frame buffer.

Assuming that the shaft encoder and external trigger signals are connected (refer to " : External Signals Connector Bracket Assembly" on page **98**), click on Grab to test the linescan camera setup. Note that CamExpert includes a **"soft" trigger** button in the Display window, which simulates an external trigger event.

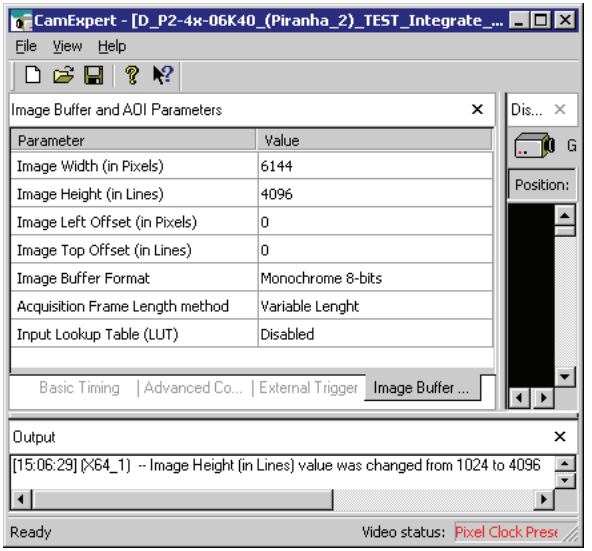
Shaft Encoder with Variable Frame Buffer Setup

To synchronize the capture of a variable amount of linescan data, external trigger signals input (Min. pulse—100ns) to the X64-CL iPro are used to start and stop line acquisition into the frame buffer. The grab control can either be a level type (where the acquisition occurs while the external control is active high), or the grab control can be a two pulse trigger (where trigger pulse 1 starts the line acquisition and trigger pulse 2 stops acquisition.

The actual linescan data capture is still triggered by the shaft encoder inputs. If the frame buffer is filled before the stop acquisition trigger control occurs, the linescan capture is suspended until the next start acquisition trigger. Example screen images of the External Trigger Parameters and the Image Buffer Parameters tabs follow.



- **External Trigger**: Set to Enable
- **External Trigger Detection**: In this example trigger input 1 starts the linescan acquisition and trigger input 2 ends the acquisition. The number of lines is variable.
- **External Trigger Level**: Choose the signal type connect as an external trigger.



- **Image Width**: By default the buffer is the same as the acquisition width.
- **Image Height**: Set to the maximum number of acquired lines needed. Refer to the hardware specifications for the maximum limit.
- **Acquisition Frame Length method**: In this example, set to variable length when there is both a start trigger pulse and an end trigger pulse to control the acquisition into the frame buffer. A level trigger signal (active high period or active low period) is also used for a grab controller with a variable length frame buffer.

Assuming that the shaft encoder and external trigger signals are connected (refer to ": External Signals Connector Bracket Assembly" on page **98**), click on Grab to test the linescan camera setup. Note that CamExpert includes a **"soft" trigger** button in the Display window, which simulates an external trigger event.

Using the Flat Field Correction Tool

Flat Field Correction is the process of eliminating small gain differences between pixels in a CCD array. That sensor when exposed to a uniformly lit field will have no gray level differences between pixels when calibrated flat field correction is applied to the image. The CamExpert Flat Field tool functions with hardware supporting flat field processing.

X64-CL iPro Flat Field Support

The X64-CL iPro supports hardware based real-time Flat Field Correction when used with its dual Base or one Medium configuration.

Flat field and flat line correction impose limitations to the maximum acquisition frame rate. Please contact the DALSA support group for more details on camera specific maximum supported acquisition rates.

Note: Sapera LT 5.2 or later is required for using hardware-based Flat Field Correction.

Loading the Required Camera File

Select the required camera configuration file for the connected camera. For a more detailed description of the server and camera file selection menu see "Using the Grab Demo" on page 53. Verify the acquisition with the live grab function. Make camera adjustments to get good images.

Also at this time make preparations to grab a flat gray level image such as a clean evenly lighted white wall or non-glossy paper. Note the lens iris position for a white but not saturated image. This white image is required for the calibration process.

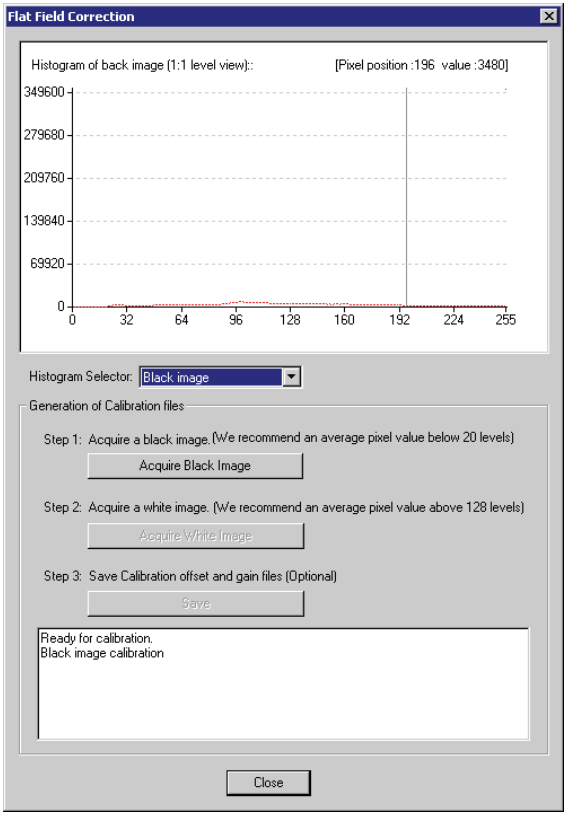
Flat Field Correction Calibration Procedure

Calibration is the process of taking two reference images, one of a black field – one of a light gray field (not saturated), to generate correction data for images captured by the CCD. Each CCD pixel data is modified by the correction factor generated by the calibration process, so that each pixel now has an identical response to the same illumination.

Start the Flat Field calibration tool via the CamExpert menu bar:
Tools • Flat Field Correction • Calibration.

Flat Field Calibration Window

The Flat Field calibration window provides a three step process to acquire two reference images and then save the flat field correction data for the camera used. To aid in determining if the reference images are valid, a histogram tool is provided so that the user can review the images used for the correction data.



- Setup the camera to capture a uniform black image. Black paper with no illumination and the camera lens' iris closed to minimum can provide such a black image.
- Click on **Acquire Black Image**. The flat field demo will grab a video frame, analyze the pixel gray level spread, and present the statistics. The desired black reference image should have pixel values less then 20. If acceptable accept the image as the black reference.
- Setup the camera to acquire a uniform white image (but not saturated white). Even illumination on white paper can be used, with a gray level of 128 minimum. It is preferable to prepare for the white level calibration before the calibration procedure.
- Click on **Acquire White Image**. The flat field demo will grab a video frame, analyze the pixel gray level spread, and present the statistics. The captured gray level for all pixels should be greater than 128. If acceptable accept the image as the white reference.
- Click on **Save**. The flat field correction data is saved as a TIF image with a file name of your choice (such as camera name and serial number).

Using Flat Field Correction

From the CamExpert menu bar enable Flat Field correction (**Tools • Flat Field Correction • Enable**). Now when doing a live grab or snap, the incoming image is corrected by the current flat field calibration data for each pixel.

Use the menu function **Tools • Flat Field Correction • Load** to load in a flat field correction image from a previous saved calibration data. CamExpert allows saving and loading calibration data for all cameras used with the imaging system.

Using the Bayer Filter Tool

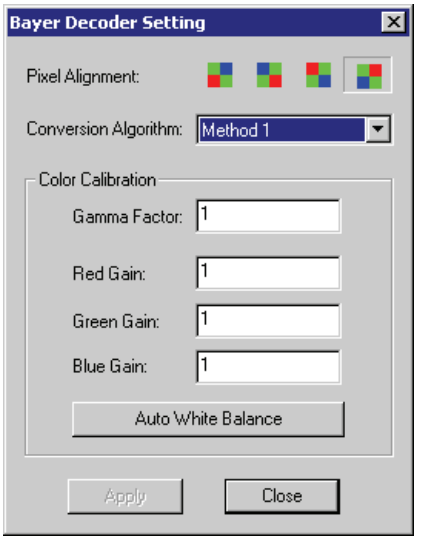
CamExpert supports the use of Bayer Filter cameras by providing a tool to select the Bayer filter mosaic pattern and to perform an auto white balance. Color calibration can then be manually fine tuned with RGB gain and gamma adjustments.

The CamExpert Bayer filter tool supports using both software or hardware based decoding. With boards that have Bayer filter decoding in hardware, such as the X64-CL iPro with its Bayer decoding configuration, CamExpert directly controls the hardware for high performance real-time acquisitions from Bayer filter cameras. When standard acquisition boards are used, CamExpert performs software Bayer filter decoding using the host system.

Bayer Filter White Balance Calibration Procedure

The following procedure uses an X64-CL iPro with hardware Bayer filter support and any Bayer color camera. It is assumed that CamExpert was used to generate a camera file with correct camera timing parameters.

- On the CamExpert menu bar, click on **Tools • Bayer Filter**. The following menu should show **Hardware** selected by default when the X64-CL iPro has Bayer support.
- Select **Setting** to access the color calibration window (see following figure).



- Click **Grab** to start live acquisition.
- Aim and focus the camera. The camera should see an area of white or place white paper in front of the object being imaged.
- Click on one of the four Bayer pixel alignment patterns to match the camera (best color before calibration). Typically the CamExpert default is correct for a majority of cameras.
- Adjust the lens iris to reduce the exposure brightness so that the white image area is now darker. Make certain that no pixel in the white area is saturated.
- Using the mouse left button, click and drag a ROI enclosing a portion of the white area.
- Click on the **Auto White Balance** button. CamExpert will make RGB gain adjustments.
- Open the camera iris to have a correctly exposed image.
- Review the image for color balance.
- Manually make additional adjustments to the RGB gain values. Fine tune the color balance to achieve best results. Adjust the gamma factor to additionally improve the display.
- Stop the live acquisition and save camera file (which now contains the Bayer RGB calibration information). Note that the gamma factor is not save because it is not a Sapera parameter but only a display tool.

Using the Bayer Filter

A Sapera application, when loading the camera file parameters, will have the RGB gain adjustment values. The application can provide the calibration window to make RGB adjustments as required.

Sapera Demo Applications

Grab Demo Overview

Program	**Start•Programs•DALSA•Sapera LT•Demos•Grab Demo**
Program file	**\DALSA\Sapera\Demos\Classes\vc\GrabDemo\Release\GrabDemo.exe**
Workspace	**\DALSA\Sapera\Demos\Classes\vc\SapDemos.dsw**
Description	This program demonstrates the basic acquisition functions included in the Sapera library. The program allows you to acquire images, either in continuous or in one-shot mode, while adjusting the acquisition parameters. The program code may be extracted for use within your own application.
Remarks	This demo is built using Visual C++ 6.0 using the MFC library. It is based on Sapera C++ classes. See the Sapera User's and Reference manuals for more information.

Using the Grab Demo

Server Selection

Run the grab demo from the start menu **Start • Programs • Sapera LT • Demos • Grab Demo**.

The demo program first displays the acquisition configuration menu. The first drop menu displayed permits selecting from any installed Sapera acquisition servers (installed DALSA acquisition hardware using Sapera drivers). The second drop menu permits selecting from the available input devices present on the selected server.

CCF File Selection

The acquisition configuration menu is also used to select the required camera configuration file for the connected camera. Sapera camera files contain timing parameters and video conditioning parameters. The default folder for camera configuration files is also used by the CamExpert utility to save user generated or modified camera files.

Use the Sapera CamExpert utility program to generate the camera configuration file based on timing and control parameters entered. The CamExpert live acquisition window allows immediate verification of those parameters. CamExpert reads both Sapera *.cca and *.cvi for backward compatibility with the original Sapera camera files.

Grab Demo Main Window

Refer to the Sapera LT User's Manual (OC-SAPM-USER), in section "Demos and Examples – Acquiring with Grab Demo", for more information on the Grab Demo.

Flat-Field Demo Overview

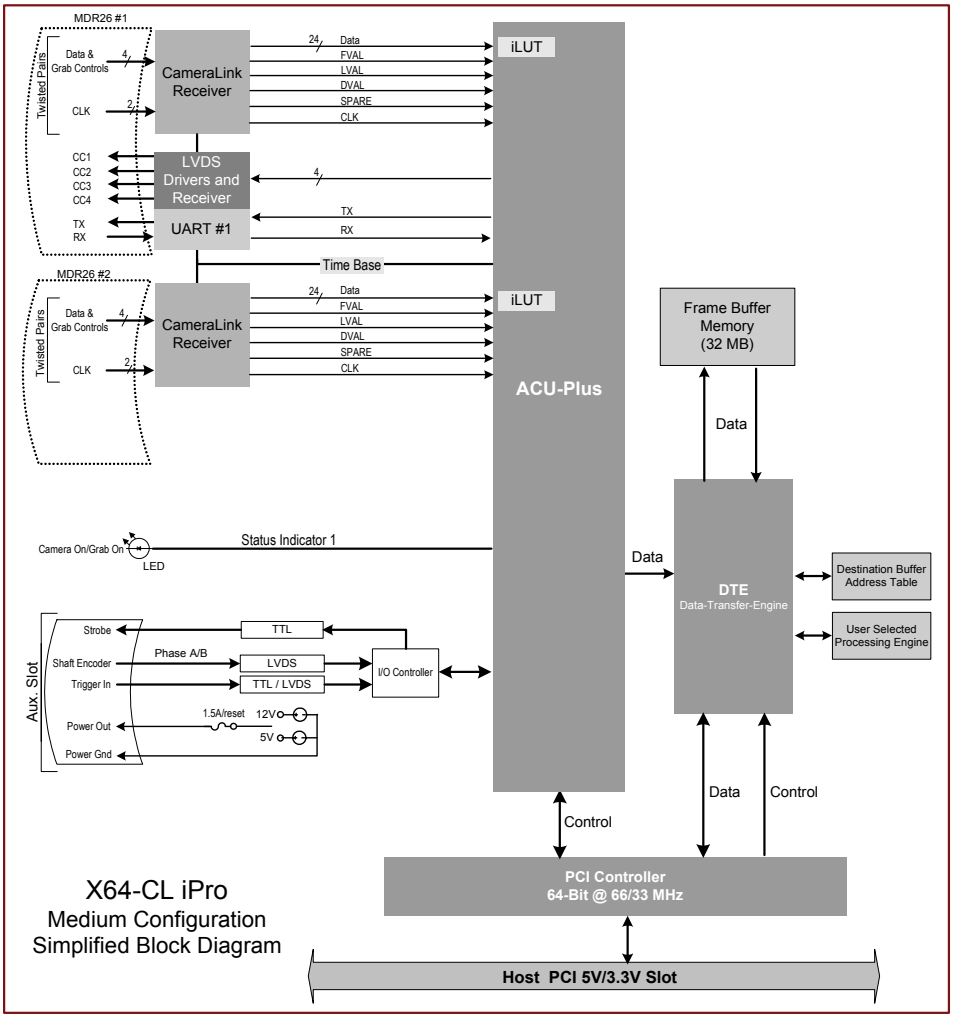
Program	**Start•Programs•DALSA•Sapera LT•Demos•Flat Field Demo**
Program file	**\dalsa\Sapera\Demos\Classes\vc\FlatFieldDemo\Release\FlatfieldDemo.exe**
Workspace	**\dalsa\Sapera\Demos\Classes\vc\SapDemos.dsw**
Description	This program demonstrates Flat Field or Flat Line processing, either performed by supporting DALSA hardware or performed on the host system via the Sapera library. The program allows you to acquire a flat field or flat line reference image, and then do real time correction either in continuous or single acquisition mode. The program code may be extracted for use within your own application.
Remarks	This demo is built using Visual C++ 6.0 using the MFC library. It is based on Sapera C++ classes. See the Sapera User's and Reference manuals for more information.

Using the Flat Field Demo

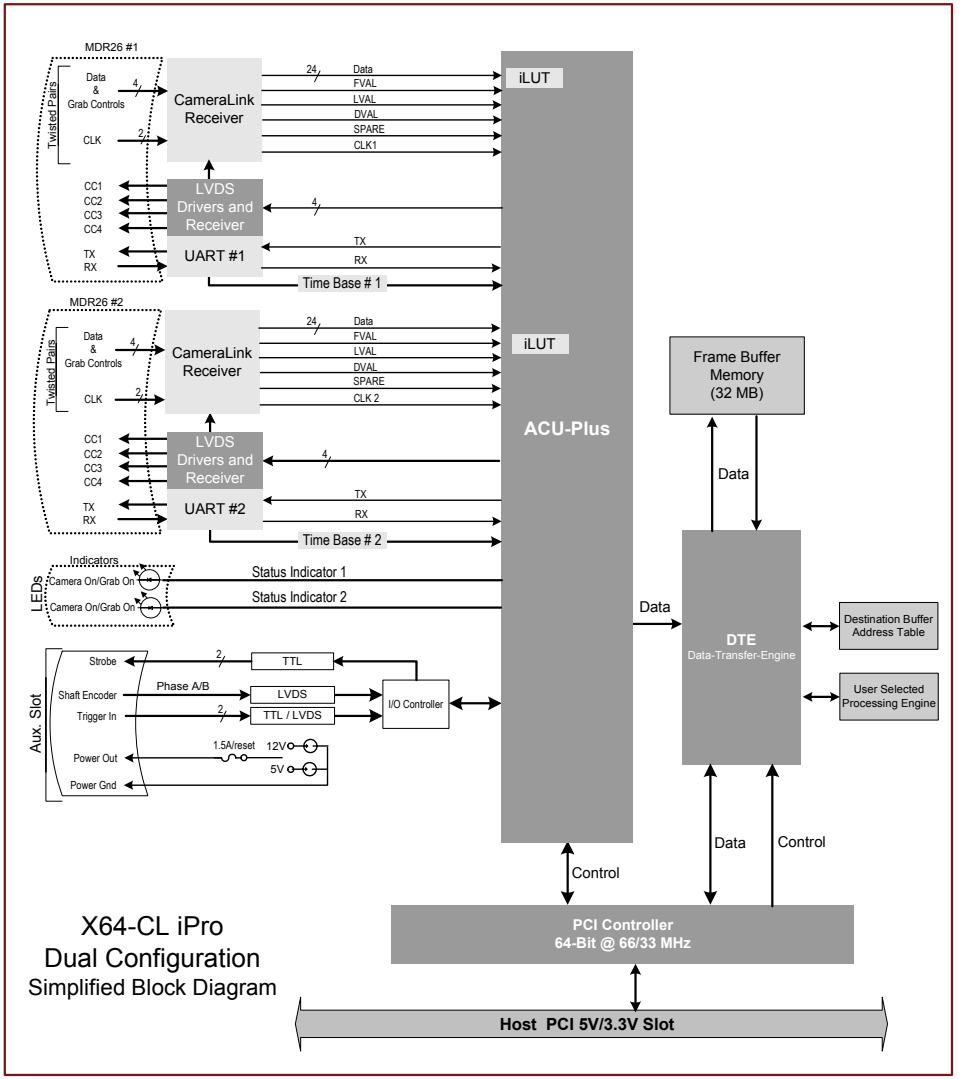
Refer to the Sapera LT User's Manual (OC-SAPM-USER), in section "Using the Flat Field Demo", for more information.

X64-CL iPro Reference

X64-CL iPro Medium Block Diagram

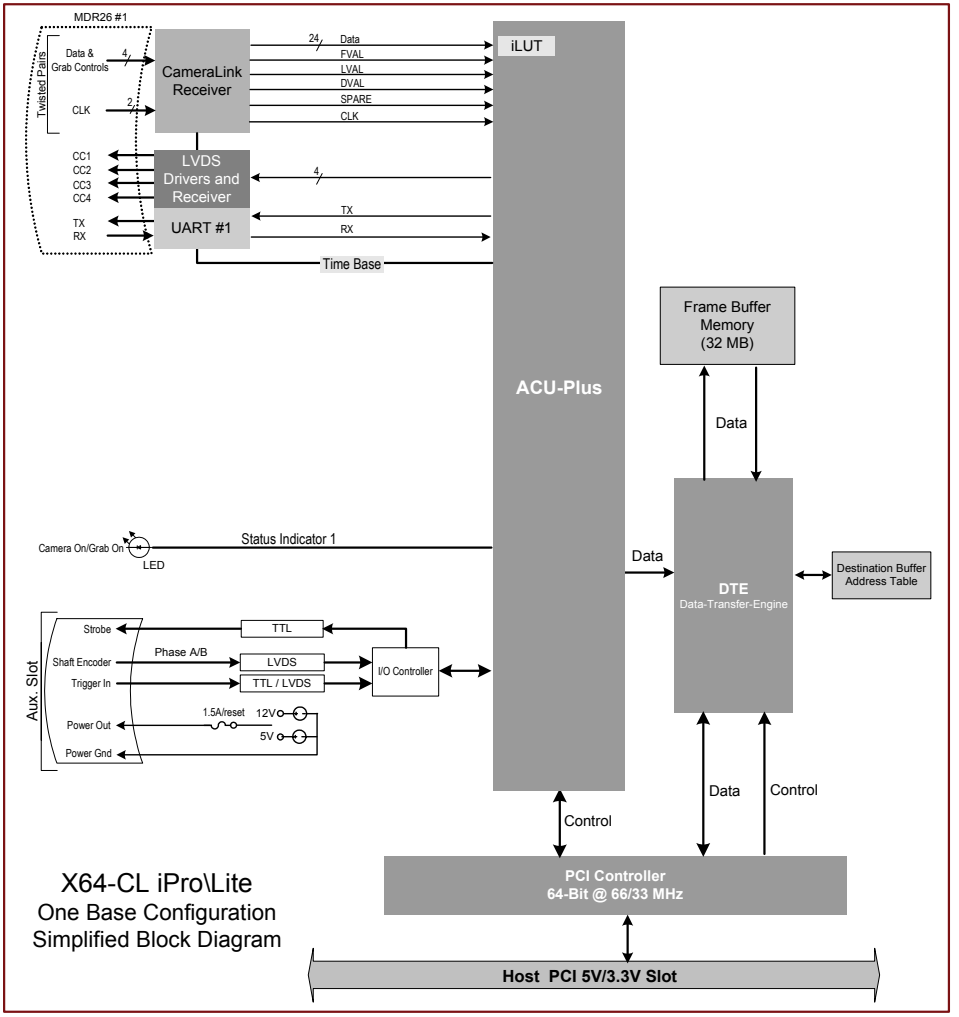


X64-CL iPro Dual Base Block Diagram

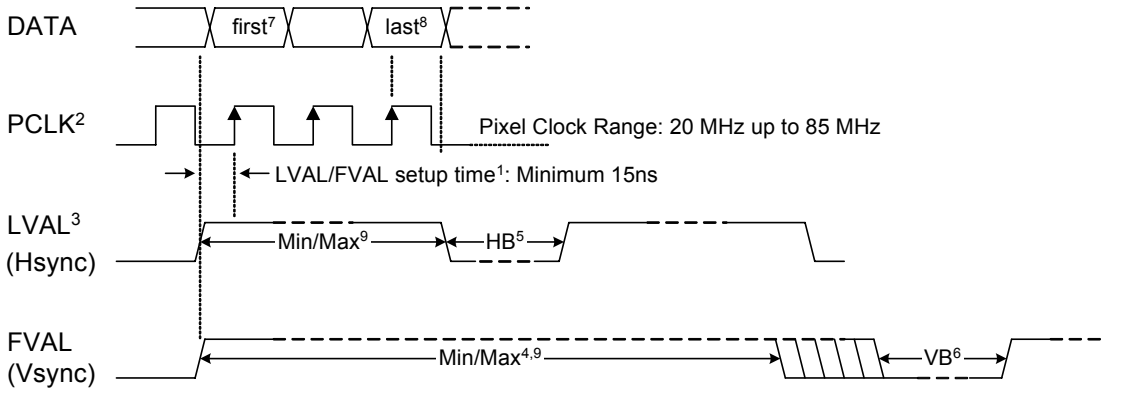


X64-CL iPro
Dual Configuration
Simplified Block Diagram

X64-CL iPro\Lite Block Diagram



X64-CL iPro Acquisition Timing



- [1] The setup times for LVAL and FVAL are the same. Both must be high and stable before the rising edge of the Pixel Clock.
- [2] Pixel Clock must always be present.
- [3] LVAL must be active high to acquire camera data.
- [4] Minimum of 1.
- [5] HB - Horizontal Blanking:

  Minimum: 4 clocks/cycle

  Maximum: no limits

- [6] VB - Vertical Blanking:

  Minimum: 1 line

  Maximum: no limits

- [7] First Active Pixel (unless otherwise specified in the CCA file – "Horizontal Back invalid = x" where 'x' defines the number of pixels to be skipped).
- [8] Last Active Pixel – defined in the CCA file under "Horizontal active = y" – where 'y' is the total number of active pixels per tap.
- [9] Maximum Valid Data:
  - 8-bits/pixel x 256K Pixels/line (LVAL)
  - 16-bits/pixel x 128K Pixels/line (LVAL)
  - 32-bits/pixel x 64K Pixels/line (LVAL)
  - 64-bits/pixel x 32K Pixels/line (LVAL)
  - 16,000,000 lines (FVAL)

Line Trigger Source Selection for Linescan Applications

Linescan imaging applications require some form of external event trigger to synchronize linescan camera exposures to the moving object. This synchronization signal is either an external trigger source (one exposure per trigger event) or a shaft encoder source composed of a single or dual phase (quadrature) signal. The X64-CL iPro shaft encoder inputs provide additional functionality with pulse drop or pulse multiply support.

The following table describes the line trigger source types supported by either the X64-CL iPro or X64-CL iPro\Lite boards. Refer to the Sapera Acquisition Parameters Reference Manual (OC-SAPM-APR00) for descriptions of the Sapera parameters.

CORACQ_PRM_EXT_LINE_TRIGGER_SOURCE – Parameter Values Specific to the X64-CL iPro series

PRM Value	Active Shaft Encoder Input
0	Default
1	Use phase A
2	Use phase B
3	Use phase A & B
4	From Board Sync
5	Phase A & B, shaft encoder after pulse drop/multiply output to Board Sync
6	Phase A & B, camera line trigger output to Board Sync

CORACQ_PRM_EXT_LINE_TRIGGER_SOURCE full description relative to trigger type and X64-CL iPro configuration used:

PRM Value	X64-CL iPro configuration & camera input used	External Line Trigger Signal used	External Shaft Encoder Signal used
		if CORACQ_PRM_EXT_LINE_ TRIGGER_ENABLE = *true*	*if* CORACQ_PRM_SHAFT_ ENCODER_ENABLE =*true*
0	Dual - Camera #1	Shaft Encoder Phase A	Shaft Encoder Phase A
	Dual - Camera #2	Shaft Encoder Phase B	Shaft Encoder Phase B
	Medium - Camera #1	Shaft Encoder Phase A	Shaft Encoder Phase A & B
1	Dual - Camera #1	Shaft Encoder Phase A	Shaft Encoder Phase A
	Dual - Camera #2	Shaft Encoder Phase A	Shaft Encoder Phase A
	Medium - Camera #1	Shaft Encoder Phase A	Shaft Encoder Phase A
2	Dual - Camera #1	Shaft Encoder Phase B	Shaft Encoder Phase B
	Dual - Camera #2	Shaft Encoder Phase B	Shaft Encoder Phase B
	Medium - Camera #1	Shaft Encoder Phase B	Shaft Encoder Phase B
3, 5, 6	Dual - Camera #1	n/a	Shaft Encoder Phase A & B
	Dual - Camera #2	n/a	Shaft Encoder Phase A & B
	Medium - Camera #1	n/a	Shaft Encoder Phase A & B
4	Dual Camera #1	From Board Sync	From Board Sync
	Dual Camera #2	From Board Sync	From Board Sync
	Medium Camera #1	From Board Sync	From Board Sync

See "J4: External Signals Connector " on page 95 for shaft encoder input connector details.

CVI/CCF File Parameters Used

- External Line Trigger Source = prm value
- External Line Trigger Enable = true/false
- Shaft Encoder Enable = true/false

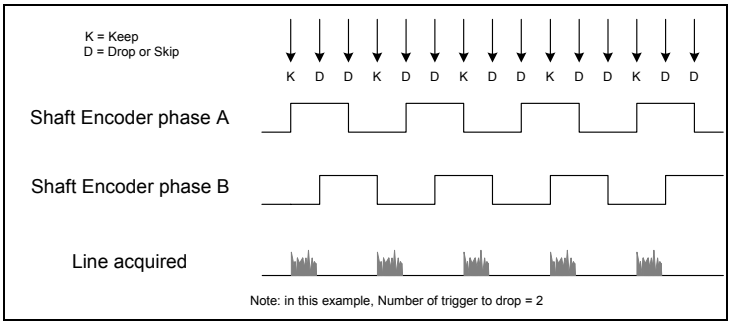
Shaft Encoder Interface Timing

Connector J4, Dual Balanced Shaft Encoder Inputs:

- Input 1: Pin 15 (Phase A +) & Pin 16 (Phase A -)
- Input 2: Pin 17 (Phase B +) & Pin 18 (Phase B -)
- (see "J4: External Signals Connector " on page 95 for complete connector signal details)
- (also see "X64-CL iPro: External Signals Connector Bracket Assembly" on page 98 for pinout)

Web inspection systems with variable web speeds typically provide one or two synchronization signals from a web mounted encoder to coordinate trigger signals. These trigger signals are used by the acquisition linescan camera. The X64-CL iPro supports single or dual shaft encoder signals. Dual encoder signals are typically 90 degrees out of phase relative to each other and provide greater web motion resolution.

When enabled, the camera is triggered and acquires one scan line for each shaft encoder pulse edge. To optimize the web application, a second Sapera parameter defines the number of triggers to skip between valid acquisition triggers. The figure below depicts a system where a valid camera trigger is any pulse edge from either shaft encoder signal. After a trigger the two following triggers are ignored (as defined by the Sapera pulse drop parameter).


K = Keep
D = Drop or Skip
K D D K D D K D D K D D K D D
Shaft Encoder phase A
Shaft Encoder phase B
Line acquired
Note: in this example, Number of trigger to drop = 2

Note that camera file parameters are best modified by using the Sapera CamExpert program.

CVI/CCF File Parameters Used

Shaft Encoder Enable = X, where:

- If X = 1, Shaft Encoder is enabled
- If X = 0, Shaft Encoder is disabled

Shaft Encoder Pulse Drop = X, where:

- X = number of trigger pulses ignored between valid triggers

For information on camera configuration files see the Sapera Acquisition Parameters Reference Manual (OC-SAPM-APR00).

Virtual Frame Trigger for Linescan Cameras

When using linescan cameras a frame buffer is allocated in host system memory to store captured video lines. To control when a video line is stored as the first line in this "virtual" frame buffer, an external frame trigger signal is used. The number of lines sequentially grabbed and stored in the virtual frame buffer is controlled by the Sapera vertical cropping parameter.

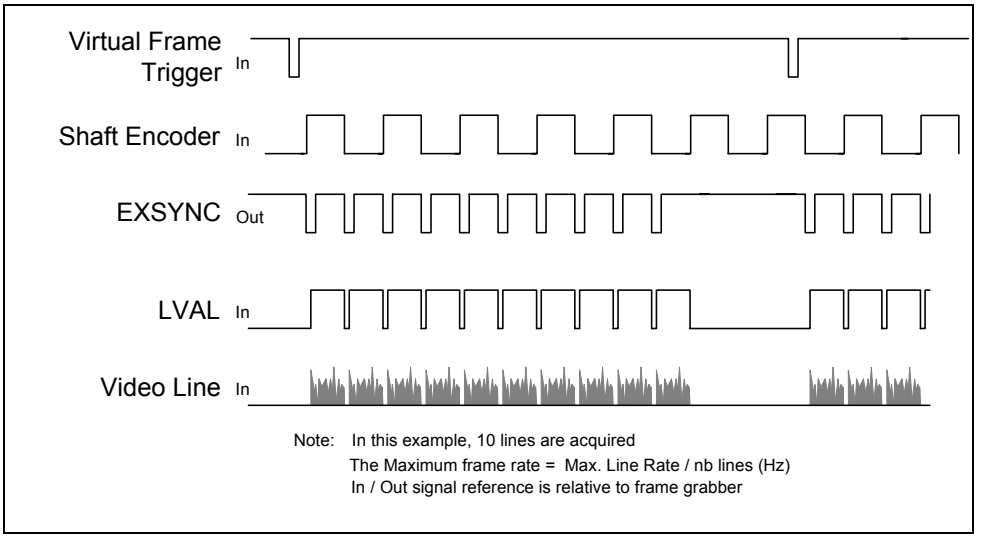
Virtual Frame Trigger Timing Diagram

The following timing diagram shows an example of grabbing 10 video lines from a linescan camera and the use of virtual frame trigger to define when a video line is stored at the beginning of the virtual frame buffer. The virtual frame trigger signal (generated by some external event) is input on the X64-CL iPro trigger input.

- **Virtual frame trigger** can be TTL 5V or LVDS and be rising or falling edge active, active high or low, or double pulse rising or falling edge.
- **Virtual frame trigger** control is configured for rising edge trigger in this example.
- **Virtual frame trigger** connects to the X64-CL iPro via the External Trigger Input 1 & 2 balanced inputs on connector J4, Input 1: pin 11 (+) and 12 (-), Input 2: pin 13 (+) and 14 (-).
- After the X64-CL iPro receives virtual frame trigger, the camera control signal is output to the camera to trigger n lines of video as per the defined virtual frame size.
- The camera control signal is either based on timing controls input on one or both X64-CL iPro shaft encoder inputs (see "J4: External Signals Connector " on page **95** pinout) or an internal X64-CL iPro clock.
- The number of lines captured is specified by the Sapera vertical cropping parameter.

Synchronization Signals for a Virtual Frame of 10 Lines.

The following timing diagram shows the relationship between external Frame Trigger input, external Shaft Encoder input (one phase used), and camera control output to the camera.


Virtual Frame Trigger In
Shaft Encoder In
EXSYNC Out
LVAL In
Video Line In
Note: In this example, 10 lines are acquired
The Maximum frame rate = Max. Line Rate / nb lines (Hz)
In / Out signal reference is relative to frame grabber

CCF File Parameters Used

The parameters listed below provide the control functionality for virtual frame reset. Applications either load pre-configured .CCF files or change these parameters directly during runtime.

Note that camera file parameters are best modified by using the Sapera CamExpert program.

External Frame Trigger Enable = X, where: \\Virtual Frame Trigger enabled

- If X = 1, External Frame Trigger is enabled
- If X = 0, External Frame Trigger is disabled

External Frame Trigger Detection = Y, where: \\ with Virtual Frame Trigger edge select

- If Y= 1, External Frame Trigger is active low
- If Y= 2, External Frame Trigger is active high
- If Y= 4, External Frame Trigger is active on rising edge
- If Y= 8, External Frame Trigger is active on falling edge
- If Y= 32, External Frame Trigger is dual-input rising edge
- If Y= 64, External Frame Trigger is dual-input falling edge

External Frame Trigger Level = Z, where: \\ with Virtual Frame Trigger signal type

- If Z= 2, External Frame Trigger is a RS-422/LVDS signal

For information on camera files see the Sapera Acquisition Parameters Reference Manual (OC-SAPM-APR00).

Acquisition Methods

Sapera acquisition methods define the control and timing of the camera and frame grabber board. Various methods are available, grouped as:

- Camera Trigger Methods (method 1 and 2 supported)
- Line Integration Methods (method 1 through 4 supported)
- Line Trigger Method 1 supported
- Time Integration Methods (method 1 through 8 supported)
- Strobe Methods (method 1 through 4 supported)

Refer to X64-CL iPro Sapera Parameters for methods supported by the X64-CL iPro. Refer to the Sapera LT Acquisition Parameters Reference manual (OC-SAPM-APR00) for detailed information concerning camera and acquisition control methods.

X64-CL iPro LUT availability

The following table defines the X64-CL iPro input LUT availability. For the X64-CL iPro\Lite see X64-CL iPro\Lite LUT availability.

Number of Digital Bits	Number of Taps Medium	Number of Taps Dual Base	Output Pixel Format	LUT Format	Notes
8	4	3	MONO 8		Yes
8	4	3	MONO 16	-	Not Supported
10	4	2	MONO 8		Yes
10	4	2	MONO 16		10 bits in 10 LSBs of 16-bit
12	4	2	MONO 8	12-in, 8-out	8 MSB
12	4	2	MONO 16	12-in, 12-out	12 bits in 12 LSBs of 16-bit
14	-	-	MONO 8	-	Not Supported
14	-	-	MONO 16	-	Not Supported
16	-	-	MONO 8	-	Not Supported
16	-	-	MONO 16	-	Not Supported
8 x 3 (RGB)	2	1	RGB8888	3 x 8 bit	Yes
8 x 3 (RGB)	1	1	RGB101010	-	Not Supported
10 x 3 (RGB)	1	-	RGB888		Medium only
10 x 3 (RGB)	1	-	RGB101010		Medium only
12 x 3 (RGB)	1	-	RGB101010		Medium Only
12 x 3 (RGB)	1	-	RGB8888		Medium Only

X64-CL iPro\Lite LUT availability

The following table defines the X64-CL iPro \Lite input LUT availability. For the X64-CL iPro see X64-CL iPro LUT availability.

Number of Digital Bits	Number of Taps	Output Pixel Format	LUT Format	Notes
8	3	MONO 8		Yes
8	3	MONO 16	-	Not Supported
10	2	MONO 8		Yes
10	2	MONO 16		10 bits in 10 LSBs of 16-bit
12	2	MONO 8	12-in, 8-out	8 MSB
12	2	MONO 16	12-in, 12-out	12 bits in 12 LSBs of 16-bit
14	-	MONO 8	-	Not Supported
14	-	MONO 16	-	Not Supported
16	-	MONO 8	-	Not Supported
16	-	MONO 16	-	Not Supported
8 x 3 (RGB)	1	RGB8888	3 x 8 bit	Yes
8 x 3 (RGB)	-	RGB101010	-	Not Supported
10 x 3 (RGB)	-	RGB888	-	Not Supported
10 x 3 (RGB)	-	RGB101010	-	Not Supported

Trigger-to-Image Reliability

Trigger-to-image reliability incorporates all stages of image acquisition inside an integrated controller to increase reliability and simplify error recovery. The trigger-to-image reliability model brings together all the requirements for image acquisition to a central management unit. These include I/O signals to control camera timing, onboard frame buffer memory to compensate for PCI bus latency, and comprehensive error notification. Whenever X64-CL iPro detects a problem, the user application is immediately informed and can take appropriate action to return to normal operation.

The X64 series is designed with a robust ACU (Acquisition and Control Unit). The ACU monitors in real-time, the acquisition state of the input plus the DTE (Data Transfer Engine), which transfers image data from on-board FIFO memory into PC memory. External trigger signal noise or glitches are easily ignored by the ACU with its programmable debounce control. The minimum pulse duration considered as a valid external trigger pulse, is set by a programmable parameter.

In general these management processes are transparent to end-user applications. Applications ensure trigger-to-image reliability by monitoring events and controlling transfer methods as described below:

Supported Events and Transfer Methods

The following acquisition and transfer events are supported. Event monitoring is a major component to the Trigger-to-Image Reliability framework.

Acquisition Events

Acquisition events are related to the acquisition module. These events provide feedback on the image capture process.

- **External Trigger** (Used/Ignored)
  Generated when the external trigger pin is asserted, usually indicating the start of the acquisition process. There are 2 types of external trigger events: 'Used' or 'Ignored'. Following an external trigger, if the event generates a captured image, an External Trigger Used event will be generated (CORACQ_VAL_EVENT_TYPE_EXTERNAL_TRIGGER and CORACQ_VAL_EVENT_TYPE_EXTERNAL_TRIGGER2).
  If there is no captured image, an External Trigger Ignored event will be generated (CORACQ_VAL_EVENT_TYPE_EXTERNAL_TRIGGER_IGNORED).
  An external trigger event will be ignored if the rate at which the events are received are higher than the possible frame rate of the camera.
- **Start of Frame**
  Event generated, during acquisition, when the video frame start is detected by the board acquisition hardware. The Sapera event value is CORACQ_VAL_EVENT_TYPE_START_OF_FRAME.
- **End of Frame**
  Event generated, during acquisition, when the video frame end is detected by the board

acquisition hardware. The Sapera event value is CORACQ_VAL_EVENT_TYPE_END_OF_FRAME.

- **Data Overflow**
  The Data Overflow event indicates that there is not enough bandwidth for the acquired data to be transferred without loss. This is usually caused by limitations of the acquisition module and should never occur.
  The Sapera event value is CORACQ_VAL_EVENT_TYPE_DATA_OVERFLOW.
- **Frame Valid**
  Event generated when the video frame start is detected by the board acquisition hardware. Acquisition does not need to be started, therefore this event can verify a valid signal is connected. The Sapera event value is CORACQ_VAL_EVENT_TYPE_VERTICAL_SYNC.
- **Pixel Clock** (Present/Absent)
  Event generated on the transition from detecting or not detecting a pixel clock signal. The Sapera event values are CORACQ_VAL_EVENT_TYPE_NO_PIXEL_CLK and CORACQ_VAL_EVENT_TYPE_PIXEL_CLK.
- **Frame Lost**
  The Frame Lost event indicates that an acquired image could not be transferred to on-board memory. An example would be if there are no free on-board buffers available for the new image. This may be the case if the image transfer from onboard buffers to host PC memory cannot be sustained due to bus bandwidth issues.
  The Sapera event value is CORACQ_VAL_EVENT_TYPE_FRAME_LOST.
- **Vertical Timeout**
  A vertical timeout event (CORACQ_VAL_EVENT_TYPE_VERTICAL_TIMEOUT) is generated when a vertical timeout is detected and calls the callback function. The vertical timeout is set using CORACQ_PRM_VERTICAL_TIMEOUT_DELAY. The vertical sync event (CORACQ_VAL_EVENT_TYPE_VERTICAL_SYNC) calls the callback function on every sync, even if not acquiring.

Transfer Events

Transfer events are the ones related to the transfer module. Transfer events provide feedback on image transfer from on-board memory frame buffers to PC memory frame buffers.

- **Start of Frame**
  The Start of Frame event is generated when the first image pixel is transferred from onboard memory into PC memory.
  The Sapera event value is CORXFER_VAL_EVENT_TYPE_START_OF_FRAME.
- **End of Frame**
  The End of Frame event is generated when the last image pixel is transferred from onboard memory into PC memory.
  The Sapera event value is CORXFER_VAL_EVENT_TYPE_END_OF_FRAME.
- **End of Line**
  The End of Line event is generated after a video line is transferred to a PC buffer.
  The Sapera event value is CORXFER_VAL_EVENT_TYPE_END_OF_LINE.

- **End of N Lines**
  The End of N Lines event is generated after a set number of video lines are transferred to a PC buffer. The Sapera event value is CORXFER_VAL_EVENT_TYPE_END_OF_NLINES.
- **End of Transfer**
  The End of Transfer event is generated at the completion of the last image being transferred from onboard memory into PC memory. To complete a transfer, a stop must be issued to the transfer module (if transfers are already in progress). If a transfer of a fixed number of frames was requested, the transfer module will stop transfers automatically.
  The Sapera event value is CORXFER_VAL_EVENT_TYPE_END_OF_TRANSFER.

Trigger Signal Validity

External trigger signal noise is easily ignored by the ACU with its programmable debounce control. A parameter is programmed for the minimum pulse duration considered as a valid external trigger pulse. Refer to "External Trigger TTL Input Electrical Specification" on page 101 for more information.

Supported Transfer Cycling Methods

The `X64-CL iPro` supports the following transfer cycle modes which are either synchronous or asynchronous. These definitions are from the Sapera Basic Reference manual.

- **CORXFER_VAL_CYCLE_MODE_SYNCHRONOUS_WITH_TRASH**
  Before cycling to the next buffer in the list, the transfer device will check the next buffer's state. If its state is full, the transfer will be done in the trash buffer which is defined as the last buffer in the list; otherwise, it will occur in the next buffer. After a transfer to the trash buffer is done, the transfer device will check again the state of the next buffer. If it is empty, it will transfer to this buffer otherwise it will transfer again to the trash buffer.
- **CORXFER_VAL_CYCLE_MODE_SYNCHRONOUS_NEXT_EMPTY_WITH_TRASH**
  Before cycling to the next buffer in the list, the transfer device will check the next buffer's state. If its state is full, the next buffer will be skipped, and the transfer will be done in the trash buffer, which is defined as the last buffer in the list; otherwise it will occur in the next buffer. After a transfer to the trash is done, the transfer device will check the next buffer in the list, if its state is empty, it will transfer to this buffer otherwise it will skip it, and transfer again to the trash buffer.
- **CORXFER_VAL_CYCLE_MODE_ASYNCHRONOUS**
  The transfer device cycles through all buffers in the list without concern about the buffer state.

X64-CL iPro Sapera Parameters

The three tables below describe the Sapera capabilities supported by the X64-CL iPro and X64-CL iPro\Lite boards. Unless specified, each capability applies to both boards or all mode configurations and all acquisition modes. Sapera parameters not used have been omitted for clarity.

The board parameter listing is subject to change. Parameters should be verified by the application because new board driver releases may change certain capabilities.

Specifically the X64-CL iPro family is described in Sapera as:

- Board Server: X64-CL_iPro_1
- Acquisition Device: Camera Link
- Modes (X64-CL iPro): Medium Mono or Medium Color RGB with Flat Field correction
- Modes (X64-CL iPro): Base Mono with Bayer mosaic filter decoding
- Modes (X64-CL iPro): Dual Base Mono or Dual Base RGB with Flat Field correction
- Mode (X64-CL iPro\Lite): One Base Mono

Capability	Values
CORACQ_CAP_CONNECTOR_TYPE	CORACQ_VAL_CONNECTOR_TYPE_CAMLINK (0x2)
CORACQ_CAP_CONNECTOR_CAMLINK (Pin – 01)	CORACQ_VAL_SIGNAL_NAME_NO_CONNECT (0x1) CORACQ_VAL_SIGNAL_NAME_PULSE0 (0x8) CORACQ_VAL_SIGNAL_NAME_PULSE1 (0x10) CORACQ_VAL_SIGNAL_NAME_GND (0x4000)
CORACQ_CAP_CONNECTOR_CAMLINK (Pin – 02)	CORACQ_VAL_SIGNAL_NAME_NO_CONNECT (0x1) CORACQ_VAL_SIGNAL_NAME_PULSE0 (0x8) CORACQ_VAL_SIGNAL_NAME_PULSE1 (0x10) CORACQ_VAL_SIGNAL_NAME_GND (0x4000)
CORACQ_CAP_CONNECTOR_CAMLINK (Pin – 03)	CORACQ_VAL_SIGNAL_NAME_NO_CONNECT (0x1) CORACQ_VAL_SIGNAL_NAME_PULSE0 (0x8) CORACQ_VAL_SIGNAL_NAME_PULSE1 (0x10) CORACQ_VAL_SIGNAL_NAME_GND (0x4000)
CORACQ_CAP_CONNECTOR_CAMLINK (Pin – 04)	CORACQ_VAL_SIGNAL_NAME_NO_CONNECT (0x1)

Camera Related Parameters

Parameter		Values
CORACQ_PRM_CHANNEL	*Medium Mono* *Base Mono* *Bayer color* *Medium RGB:* *Base RGB:*	1, 2, 3 1, 2, 3 1 1 1
CORACQ_PRM_FRAME		CORACQ_VAL_FRAME_PROGRESSIVE (0x2)
CORACQ_PRM_INTERFACE		CORACQ_VAL_INTERFACE_DIGITAL (0x2)
CORACQ_PRM_SCAN	*All* *All except Bayer color*	CORACQ_VAL_SCAN_AREA (0x1) CORACQ_VAL_SCAN_LINE (0x2)
CORACQ_PRM_SIGNAL		CORACQ_VAL_SIGNAL_DIFFERENTIAL (0x2)
CORACQ_PRM_VIDEO	*All Mono and Bayer* *All Color RGB:*	CORACQ_VAL_VIDEO_MONO (0x1) CORACQ_VAL_VIDEO_RGB (0x8)
CORACQ_PRM_PIXEL_DEPTH		

	Medium, Base Mono:	8 bits, CORDATA_FORMAT_MONO8 LUT=1 10 bits, CORDATA_FORMAT_MONO8 LUT=1 10 bits, CORDATA_FORMAT_MONO10 LUT=1 12 bits, CORDATA_FORMAT_MONO8 LUT=1 12 bits, CORDATA_FORMAT_MONO12 LUT=1 14 bits, CORDATA_FORMAT_MONO14 LUT=0 16 bits, CORDATA_FORMAT_MONO16 LUT=0
	Bayer color	8 bits, CORDATA_FORMAT_COLORNI8 LUT=1 10 bits, CORDATA_FORMAT_COLORNI10 LUT=1 12 bits, CORDATA_FORMAT_COLORNI12 LUT=1
	Base RGB	8 bits, CORDATA_FORMAT_COLORNI8 LUT=1
	Medium RGB:	8 bits, CORDATA_FORMAT_COLORNI8 LUT=1 10 bits, CORDATA_FORMAT_COLORNI10 LUT=1 12 bits, CORDATA_FORMAT_COLORNI12 LUT=1
CORACQ_PRM_VIDEO_STD		CORACQ_VAL_VIDEO_STD_NON_STD (0x1)
CORACQ_PRM_FIELD_ORDER		CORACQ_VAL_FIELD_ORDER_NEXT_FIELD (0x4)
CORACQ_PRM_HACTIVE		min = 1 pixel max = 16777215 pixel step = 1 pixel
CORACQ_PRM_HACTIVE	*Bayer color only*	min = 1 pixel max = 8192 pixel step = 1 pixel
CORACQ_PRM_HSYNC		min = 4 pixel max = 4294967295 pixel step = 1 pixel
CORACQ_PRM_VACTIVE		min = 1 line max = 16777215 line step = 1 line
CORACQ_PRM_VSYNC		min = 0 line max = 4294967295 line step = 1 line
CORACQ_PRM_HFRONT_INVALID		min = 0 pixel max = 16777215 pixel step = 1 pixel
CORACQ_PRM_HBACK_INVALID		min = 0 pixel max = 16777215 pixel step = 1 pixel
CORACQ_PRM_VFRONT_INVALID		min = 0 line max = 16777215 line step = 1 line
CORACQ_PRM_VBACK_INVALID		min = 0 line max = 16777215 line step = 1 line
CORACQ_PRM_PIXEL_CLK_SRC		CORACQ_VAL_PIXEL_CLK_SRC_EXT (0x2)
CORACQ_PRM_PIXEL_CLK_INT		min = 20000000 Hz max = 85000000 Hz step = 1 Hz
CORACQ_PRM_PIXEL_CLK_EXT		min = 20000000 Hz max = 85000000 Hz step = 1 Hz
CORACQ_PRM_PIXEL_CLK_11		20000000 Hz
CORACQ_PRM_SYNC		CORACQ_VAL_SYNC_SEP_SYNC (0x4)
CORACQ_PRM_HSYNC_POLARITY		CORACQ_VAL_ACTIVE_LOW (0x1)
CORACQ_PRM_VSYNC_POLARITY		CORACQ_VAL_ACTIVE_LOW (0x1)

CORACQ_PRM_FRAME_INTEGRATE_METHOD		Not available
CORACQ_PRM_FRAME_INTEGRATE_POLARITY		Not available
CORACQ_PRM_TIME_INTEGRATE_METHOD		CORACQ_VAL_TIME_INTEGRATE_METHOD_1 (0x1) CORACQ_VAL_TIME_INTEGRATE_METHOD_2 (0x2) CORACQ_VAL_TIME_INTEGRATE_METHOD_3 (0x4) CORACQ_VAL_TIME_INTEGRATE_METHOD_4 (0x8) CORACQ_VAL_TIME_INTEGRATE_METHOD_5 (0x10) CORACQ_VAL_TIME_INTEGRATE_METHOD_6 (0x20) CORACQ_VAL_TIME_INTEGRATE_METHOD_7 (0x40) CORACQ_VAL_TIME_INTEGRATE_METHOD_8 (0x80) CORACQ_VAL_TIME_INTEGRATE_METHOD_9 (0x100)
CORACQ_PRM_CAM_TRIGGER_METHOD		CORACQ_VAL_CAM_TRIGGER_METHOD_1 (0x1) CORACQ_VAL_CAM_TRIGGER_METHOD_2 (0x2)
CORACQ_PRM_CAM_TRIGGER_POLARITY		CORACQ_VAL_ACTIVE_LOW (0x1) CORACQ_VAL_ACTIVE_HIGH (0x2)
CORACQ_PRM_CAM_TRIGGER_DURATION		min = 1 µs max = 65535000 µs step = 1 µs
CORACQ_PRM_CAM_RESET_METHOD		CORACQ_VAL_CAM_RESET_METHOD_1 (0x1)
CORACQ_PRM_CAM_RESET_POLARITY		CORACQ_VAL_ACTIVE_LOW (0x1) CORACQ_VAL_ACTIVE_HIGH (0x2)
CORACQ_PRM_CAM_RESET_DURATION		min = 1 µs max = 65535000 µs step = 1 µs
CORACQ_PRM_CAM_NAME		Default Area Scan
CORACQ_PRM_LINE_INTEGRATE_METHOD		CORACQ_VAL_LINE_INTEGRATE_METHOD_1 (0x1) CORACQ_VAL_LINE_INTEGRATE_METHOD_2 (0x2) CORACQ_VAL_LINE_INTEGRATE_METHOD_3 (0x4) CORACQ_VAL_LINE_INTEGRATE_METHOD_4 (0x8) CORACQ_VAL_LINE_INTEGRATE_METHOD_7 (0x40)
CORACQ_PRM_LINE_TRIGGER_METHOD		CORACQ_VAL_LINE_TRIGGER_METHOD_1 (0x1)
CORACQ_PRM_LINE_TRIGGER_POLARITY		CORACQ_VAL_ACTIVE_LOW (0x1) CORACQ_VAL_ACTIVE_HIGH (0x2)
CORACQ_PRM_LINE_TRIGGER_DELAY		min = 0 µs max = 65535 µs step = 1 µs
CORACQ_PRM_LINE_TRIGGER_DURATION		min = 0 µs max = 65535 µs step = 1 µs
CORACQ_PRM_TAPS	*Medium Mono:* *Base Mono:* *Medium RGB:* *Base RGB:* *Bayer Color*	max = 6 taps max = 3 taps max = 4 taps max = 1 tap max = 3 taps
CORACQ_PRM_TAP_OUTPUT	*All modes* *All modes* Medium mono and Bayer color *only*	CORACQ_VAL_TAP_OUTPUT_ALTERNATE (0x1) CORACQ_VAL_TAP_OUTPUT_SEGMENTED (0x2) CORACQ_VAL_TAP_OUTPUT_PARALLEL (0x4)
CORACQ_PRM_TAP_1_DIRECTION	*All mono and color versions*	CORACQ_VAL_TAP_DIRECTION_LR (0x1) CORACQ_VAL_TAP_DIRECTION_RL (0x2) CORACQ_VAL_TAP_DIRECTION_UD (0x4) CORACQ_VAL_TAP_DIRECTION_DU (0x8) CORACQ_VAL_TAP_DIRECTION_FROM_TOP (0x10) CORACQ_VAL_TAP_DIRECTION_FROM_MID (0x20) CORACQ_VAL_TAP_DIRECTION_FROM_BOT (0x40)

CORACQ_PRM_TAP_2_DIRECTION	*Base & Medium mono Medium RGB, and Bayer Color only*	CORACQ_VAL_TAP_DIRECTION_LR (0x1) CORACQ_VAL_TAP_DIRECTION_RL (0x2) CORACQ_VAL_TAP_DIRECTION_UD (0x4) CORACQ_VAL_TAP_DIRECTION_DU (0x8) CORACQ_VAL_TAP_DIRECTION_FROM_TOP (0x10) CORACQ_VAL_TAP_DIRECTION_FROM_MID (0x20) CORACQ_VAL_TAP_DIRECTION_FROM_BOT (0x40)
CORACQ_PRM_TAP_3_DIRECTION	*Base & Medium mono Medium RGB, and Bayer Color only*	CORACQ_VAL_TAP_DIRECTION_LR (0x1) CORACQ_VAL_TAP_DIRECTION_RL (0x2) CORACQ_VAL_TAP_DIRECTION_UD (0x4) CORACQ_VAL_TAP_DIRECTION_DU (0x8) CORACQ_VAL_TAP_DIRECTION_FROM_TOP (0x10) CORACQ_VAL_TAP_DIRECTION_FROM_MID (0x20) CORACQ_VAL_TAP_DIRECTION_FROM_BOT (0x40)
CORACQ_PRM_TAP_4_DIRECTION	*Medium mono and Medium RGB only*	CORACQ_VAL_TAP_DIRECTION_LR (0x1) CORACQ_VAL_TAP_DIRECTION_RL (0x2) CORACQ_VAL_TAP_DIRECTION_UD (0x4) CORACQ_VAL_TAP_DIRECTION_DU (0x8) CORACQ_VAL_TAP_DIRECTION_FROM_TOP (0x10) CORACQ_VAL_TAP_DIRECTION_FROM_MID (0x20) CORACQ_VAL_TAP_DIRECTION_FROM_BOT (0x40)
CORACQ_PRM_TAP_5_DIRECTION	*Medium mono only*	CORACQ_VAL_TAP_DIRECTION_LR (0x1) CORACQ_VAL_TAP_DIRECTION_RL (0x2) CORACQ_VAL_TAP_DIRECTION_UD (0x4) CORACQ_VAL_TAP_DIRECTION_DU (0x8) CORACQ_VAL_TAP_DIRECTION_FROM_TOP (0x10) CORACQ_VAL_TAP_DIRECTION_FROM_MID (0x20) CORACQ_VAL_TAP_DIRECTION_FROM_BOT (0x40)
CORACQ_PRM_TAP_6_DIRECTION	*Medium mono only*	CORACQ_VAL_TAP_DIRECTION_LR (0x1) CORACQ_VAL_TAP_DIRECTION_RL (0x2) CORACQ_VAL_TAP_DIRECTION_UD (0x4) CORACQ_VAL_TAP_DIRECTION_DU (0x8) CORACQ_VAL_TAP_DIRECTION_FROM_TOP (0x10) CORACQ_VAL_TAP_DIRECTION_FROM_MID (0x20) CORACQ_VAL_TAP_DIRECTION_FROM_BOT (0x40)
CORACQ_PRM_PIXEL_CLK_DETECTION		CORACQ_VAL_RISING_EDGE (0x4)
CORACQ_PRM_CHANNELS_ORDER	*Base & Medium mono*	CORACQ_VAL_CHANNELS_ORDER_NORMAL (0x1) CORACQ_VAL_CHANNELS_ORDER_REVERSE (0x2)
	Base & Medium RGB and Bayer Color	CORACQ_VAL_CHANNELS_ORDER_NORMAL (0x1)
CORACQ_PRM_LINESCAN_DIRECTION		Not available
CORACQ_PRM_LINESCAN_DIRECTION_POLARITY		Not available
CORACQ_PRM_CAM_LINE_TRIGGER_FREQ_MIN		1 Hz
CORACQ_PRM_CAM_LINE_TRIGGER_FREQ_MAX		16777215 Hz
CORACQ_PRM_CAM_TIME_INTEGRATE_DURATION_MIN		1 µs
CORACQ_PRM_CAM_TIME_INTEGRATE_DURATION_MAX		65535000 µs
CORACQ_PRM_TIME_INTEGRATE_PULSE1_POLARITY		CORACQ_VAL_ACTIVE_LOW (0x1) CORACQ_VAL_ACTIVE_HIGH (0x2)
CORACQ_PRM_TIME_INTEGRATE_PULSE1_DELAY		min = 0 µs max = 65535000 µs step = 1 µs
CORACQ_PRM_TIME_INTEGRATE_PULSE1_DURATION		min = 0 µs max = 65535000 µs step = 1 µs
CORACQ_PRM_TIME_INTEGRATE_PULSE0_POLARITY		CORACQ_VAL_ACTIVE_LOW (0x1) CORACQ_VAL_ACTIVE_HIGH (0x2)

CORACQ_PRM_TIME_INTEGRATE_PULSE0_DELAY	min = 0 µs max = 65535000 µs step = 1 µs
CORACQ_PRM_TIME_INTEGRATE_PULSE0_DURATION	min = 1 µs max = 65535000 µs step = 1 µs
CORACQ_PRM_LINE_INTEGRATE_PULSE1_POLARITY	CORACQ_VAL_ACTIVE_LOW (0x1) CORACQ_VAL_ACTIVE_HIGH (0x2)
CORACQ_PRM_LINE_INTEGRATE_PULSE1_DELAY	min = 0 µs max = 65535000 µs step = 1 µs
CORACQ_PRM_LINE_INTEGRATE_PULSE1_DURATION	min = 1 µs max = 65535000 µs step = 1 µs
CORACQ_PRM_LINE_INTEGRATE_PULSE0_POLARITY	CORACQ_VAL_ACTIVE_LOW (0x1) CORACQ_VAL_ACTIVE_HIGH (0x2)
CORACQ_PRM_LINE_INTEGRATE_PULSE0_DELAY	min = 0 µs max = 65535 µs step = 1 µs
CORACQ_PRM_LINE_INTEGRATE_PULSE0_DURATION	min = 1 µs max = 65535000 µs step = 1 µs
CORACQ_PRM_CONNECTOR_LINE_INTEGRATE_INPUT	Connector #1, type 2, pin #1
CORACQ_PRM_CAMLINK_CONFIGURATION	
Medium Mono	CORACQ_VAL_CAMLINK_CONFIGURATION_BASE (0x1) CORACQ_VAL_CAMLINK_CONFIGURATION_MEDIUM (0x2) CORACQ_VAL_CAMLINK_CONFIGURATION_2BASE (0x8)
Medium RGB	CORACQ_VAL_CAMLINK_CONFIGURATION_BASE (0x1) CORACQ_VAL_CAMLINK_CONFIGURATION_MEDIUM (0x2)
Base Mono, Base RGB, and Bayer Color	CORACQ_VAL_CAMLINK_CONFIGURATION_BASE (0x1)
CORACQ_PRM_DATA_VALID_ENABLE	TRUE / FALSE
CORACQ_PRM_DATA_VALID_POLARITY	CORACQ_VAL_ACTIVE_HIGH (0x2)
CORACQ_PRM_TIMESLOT	
Medium Mono, Medium RGB, Bayer Color	CORACQ_VAL_TIMESLOT_1 (0x1) CORACQ_VAL_TIMESLOT_2 (0x2)
Base mono, Base RGB	CORACQ_VAL_TIMESLOT_1 (0x1)
CORACQ_PRM_BAYER_ALIGNMENT *Bayer Color*	CORACQ_VAL_BAYER_ALIGNMENT_GR_BG (0x8)
All other modes	Not available
CORACQ_PRM_CAM_CONTROL_DURING_READOUT	TRUE / FALSE

VIC Related Parameters

Parameter		Values
CORACQ_PRM_CAMSEL	*all mono modes and Bayer:*	CAMSEL_MONO = from 0 to 0
	all RGB modes:	CAMSEL_RGB = from 0 to 0
CORACQ_PRM_PIXEL_MASK		Not available
CORACQ_PRM_CROP_LEFT		min = 0 pixel max = 16777215 pixel step = 8 pixel
CORACQ_PRM_CROP_TOP		min = 0 line max = 16777215 line step = 1 line
	Bayer mode only	step = 2 lines
CORACQ_PRM_CROP_WIDTH		min = 8 pixel max = 16777215 pixel step = 8 pixel
CORACQ_PRM_CROP_HEIGHT		min = 1 line max = 16777215 line step = 1 line
CORACQ_PRM_SCALE_HORZ		Not available
CORACQ_PRM_SCALE_VERT		Not available
CORACQ_PRM_DECIMATE_METHOD		CORACQ_VAL_DECIMATE_DISABLE (0x1)
CORACQ_PRM_DECIMATE_COUNT		Default = 0
CORACQ_PRM_LUT_ENABLE		True / False
CORACQ_PRM_LUT_NUMBER		Default = 0
CORACQ_PRM_STROBE_ENABLE		TRUE / FALSE
CORACQ_PRM_STROBE_METHOD		CORACQ_VAL_STROBE_METHOD_1 (0x1) CORACQ_VAL_STROBE_METHOD_2 (0x2) CORACQ_VAL_STROBE_METHOD_3 (0x4) CORACQ_VAL_STROBE_METHOD_4 (0x8)
CORACQ_PRM_STROBE_POLARITY		CORACQ_VAL_ACTIVE_LOW (0x1) CORACQ_VAL_ACTIVE_HIGH (0x2)
CORACQ_PRM_STROBE_DURATION		min = 0 µs max = 65535000 µs step = 1 µs
CORACQ_PRM_STROBE_DELAY		min = 0 µs max = 65535000 µs step = 1 µs
CORACQ_PRM_FRAME_INTEGRATE_ENABLE		Not available
CORACQ_PRM_FRAME_INTEGRATE_COUNT		Not available
CORACQ_PRM_TIME_INTEGRATE_ENABLE		TRUE / FALSE
CORACQ_PRM_TIME_INTEGRATE_DURATION		min = 1 µs max = 65535000 µs step = 1 µs
CORACQ_PRM_CAM_TRIGGER_ENABLE		TRUE / FALSE
CORACQ_PRM_CAM_RESET_ENABLE		TRUE / FALSE
CORACQ_PRM_OUTPUT_FORMAT		

	Basel mono & Medium mono & Bayer color	CORACQ_VAL_OUTPUT_FORMAT_MONO8 CORACQ_VAL_OUTPUT_FORMAT_MONO16
	Medium RGB & Bayer color	CORACQ_VAL_OUTPUT_FORMAT_RGB8888 CORACQ_VAL_OUTPUT_FORMAT_RGB101010
	Base RGB	CORACQ_VAL_OUTPUT_FORMAT_RGB8888
CORACQ_PRM_EXT_TRIGGER_ENABLE		CORACQ_VAL_EXT_TRIGGER_OFF (0x1) CORACQ_VAL_EXT_TRIGGER_ON (0x8)
CORACQ_PRM_VIC_NAME		Default Area Scan
CORACQ_PRM_LUT_MAX		1
CORACQ_PRM_EXT_TRIGGER_DETECTION		CORACQ_VAL_ACTIVE_LOW (0x1) CORACQ_VAL_ACTIVE_HIGH (0x2) CORACQ_VAL_RISING_EDGE (0x4) CORACQ_VAL_FALLING_EDGE (0x8)
CORACQ_PRM_LUT_FORMAT	*Mono mode*	Default = CORACQ_VAL_OUTPUT_FORMAT_MONO8
	Base RGB mode	Default = CORACQ_VAL_OUTPUT_FORMAT_RGB8888
	Medium RGB mode	Default = CORACQ_VAL_OUTPUT_FORMAT_RGB101010
	Bayer Color mode	Default = CORACQ_VAL_OUTPUT_FORMAT_MONO16
CORACQ_PRM_VSYNC_TIMEOUT		Not available
CORACQ_PRM_VSYNC_REF		CORACQ_VAL_SYNC_REF_END (0x2)
CORACQ_PRM_HSYNC_REF		CORACQ_VAL_SYNC_REF_END (0x2)
CORACQ_PRM_LINE_INTEGRATE_ENABLE		TRUE / FALSE
CORACQ_PRM_LINE_INTEGRATE_DURATION		min = 1 pixel max = 16777215 pixel step = 1 pixel
CORACQ_PRM_LINE_TRIGGER_ENABLE		TRUE / FALSE
CORACQ_PRM_EXT_FRAME_TRIGGER_ENABLE		TRUE / FALSE
CORACQ_PRM_EXT_FRAME_TRIGGER_DETECTION		CORACQ_VAL_ACTIVE_LOW (0x1) CORACQ_VAL_ACTIVE_HIGH (0x2) CORACQ_VAL_RISING_EDGE (0x4) CORACQ_VAL_FALLING_EDGE (0x8) CORACQ_VAL_DOUBLE_PULSE_RISING_EDGE (0x20) CORACQ_VAL_DOUBLE_PULSE_FALLING_EDGE (0x40)
CORACQ_PRM_EXT_LINE_TRIGGER_ENABLE		TRUE / FALSE
CORACQ_PRM_EXT_LINE_TRIGGER_DETECTION		CORACQ_VAL_RISING_EDGE (0x4)
CORACQ_PRM_SNAP_COUNT		min = 1 max = 65535 step = 1
CORACQ_PRM_INT_LINE_TRIGGER_ENABLE		TRUE / FALSE
CORACQ_PRM_INT_LINE_TRIGGER_FREQ		Default = 5000 Hz
CORACQ_PRM_LINESCAN_DIRECTION_OUTPUT (*)		Not available
CORACQ_PRM_BIT_ORDERING		CORACQ_VAL_BIT_ORDERING_STD (0x1)
CORACQ_PRM_EXT_TRIGGER_LEVEL		CORACQ_VAL_LEVEL_TTL (0x1) CORACQ_VAL_LEVEL_422 (0x2)
CORACQ_PRM_STROBE_LEVEL		CORACQ_VAL_LEVEL_TTL (0x1)
CORACQ_PRM_EXT_FRAME_TRIGGER_LEVEL		CORACQ_VAL_LEVEL_TTL (0x1) CORACQ_VAL_LEVEL_422 (0x2)
CORACQ_PRM_EXT_LINE_TRIGGER_LEVEL		CORACQ_VAL_LEVEL_422 (0x2)
CORACQ_PRM_INT_LINE_TRIGGER_FREQ_MIN		245 Hz
CORACQ_PRM_INT_LINE_TRIGGER_FREQ_MAX		500000 Hz

CORACQ_PRM_SHAFT_ENCODER_DROP		min = 0 tick max = 255 tick step = 1 tick
CORACQ_PRM_SHAFT_ENCODER_ENABLE		TRUE / FALSE
CORACQ_PRM_EXT_TRIGGER_FRAME_COUNT		min = 1 max = 65535 step = 1
CORACQ_PRM_INT_FRAME_TRIGGER_ENABLE		TRUE / FALSE
CORACQ_PRM_INT_FRAME_TRIGGER_FREQ		min = 1 milli-Hz max = 1073741823 milli-Hz step = 1 milli-Hz
CORACQ_PRM_STROBE_DELAY_2		min = 0 µs max = 65535000 µs step = 1 µs
CORACQ_PRM_FRAME_LENGTH		CORACQ_VAL_FRAME_LENGTH_FIX (0x1) CORACQ_VAL_FRAME_LENGTH_VARIABLE (0x2)
CORACQ_PRM_FLIP		CORACQ_VAL_FLIP_HORZ (0x1) *(not available in Bayer Color mode)*
CORACQ_PRM_EXT_TRIGGER_DURATION		min = 0 µs max = 255 µs step = 1 µs
CORACQ_PRM_TIME_INTEGRATE_DELAY		min = 0 µs max = 65535000 µs step = 1 µs
CORACQ_PRM_CAM_TRIGGER_DELAY		min = 0 µs max = 65535000 µs step = 1 µs
CORACQ_PRM_SHAFT_ENCODER_LEVEL		CORACQ_VAL_LEVEL_422 (0x2)
CORACQ_PRM_LUT_NENTRIES		256 entries
CORACQ_PRM_EXT_FRAME_TRIGGER_SOURCE		min = 0 max = 5 step = 1 See CORACQ_PRM_EXT_TRIGGER_SOURCE_STR
CORACQ_PRM_EXT_LINE_TRIGGER_SOURCE (*)		min = 0 max = 7 step = 1 See CORACQ_PRM_EXT_LINE_TRIGGER_SOURCE_STR
CORACQ_PRM_EXT_TRIGGER_SOURCE		min = 0 max = 5 step = 1 See CORACQ_PRM_EXT_TRIGGER_SOURCE_STR
CORACQ_PRM_SHAFT_ENCODER_MULTIPLY		min = 1 max = 32 step = (2**N)
CORACQ_PRM_EXT_TRIGGER_DELAY		min = 0 max = 65535000 step = 1
CORACQ_PRM_EXT_TRIGGER_DELAY_TIME_BASE		CORACQ_VAL_TIME_BASE_LINE (0x4)
CORACQ_PRM_BAYER_DECODER_ENABLE	*all modes*	Not available
	Bayer Color	True / False
CORACQ_PRM_BAYER_DECODER_METHOD	*all modes*	Not available
	Bayer color	CORACQ_VAL_BAYER_DECODER_METHOD_1 (0x1)

CORACQ_PRM_BAYER_DECODER_WB_GAIN	*Bayer color only*	min = 100000 max = 899609 step = 1
CORACQ_PRM_BAYER_DECODER_WB_GAIN_RED	*Bayer color only*	min = 100000 max = 499609 step = 1
CORACQ_PRM_BAYER_DECODER_WB_GAIN_GREEN	*Bayer color only*	min = 100000 max = 499609 step = 1
CORACQ_PRM_BAYER_DECODER_WB_GAIN_BLUE	*Bayer color only*	min = 100000 max = 899609 step = 1
CORACQ_PRM_EXT_TRIGGER_IGNORE_DELAY		min = 0 max = 65535000 step = 1
CORACQ_PRM_EXT_TRIGGER_SOURCE_STR		[0] = Automatic [1] = From External Trigger #1 [2] = From External Trigger #2 [3] = From Board Sync [4] = To Board Sync [5] = Pulse to Board Sync
CORACQ_PRM_EXT_LINE_TRIGGER_SOURCE_STR		[0] = Automatic [1] = From Shaft Encoder Phase A [2] = From Shaft Encoder Phase B [3] = From Shaft Encoder Phase A & B [4] = From Board Sync [5] = To Board Sync [6] = Pulse to Board Sync [7] = To Board Sync When Grabbing
CORACQ_PRM_VERTICAL_TIMEOUT_DELAY		min = 0 max = 16383000 step = 1
CORACQ_PRM_BAYER_DECODER_SATURATION_FACTOR	*Bayer color only*	min = 0 max = 65535 step = 1
CORACQ_PRM_BAYER_DECODER_SATURATION_WEIGHT_RED	*Bayer color only*	min = 0 max = 65535 step = 1
CORACQ_PRM_BAYER_DECODER_SATURATION_WEIGHT_GREEN	*Bayer color only*	min = 0 max = 65535 step = 1
CORACQ_PRM_BAYER_DECODER_SATURATION_WEIGHT_BLUE	*Bayer color only*	min = 0 max = 65535 step = 1
CORACQ_PRM_POCL_ENABLE		Not available
CORACQ_PRM_CROP_ACTIVATION		Not available

ACQ Related Parameters

Parameter	Values
CORACQ_PRM_LABEL	
Medium mono	CameraLink Medium Mono #1
Base mono	CameraLink Base Mono #1
Medium Color RGB	CameraLink Medium Color RGB #1
Base Color RGB	CameraLink Base Color RGB #1
Bayer color	CameraLink Bayer #1
CORACQ_PRM_EVENT_TYPE	CORACQ_VAL_EVENT_TYPE_START_OF_FRAME CORACQ_VAL_EVENT_TYPE_END_OF_FRAME CORACQ_VAL_EVENT_TYPE_EXTERNAL_TRIGGER CORACQ_VAL_EVENT_TYPE_VERTICAL_SYNC CORACQ_VAL_EVENT_TYPE_NO_PIXEL_CLK CORACQ_VAL_EVENT_TYPE_PIXEL_CLK CORACQ_VAL_EVENT_TYPE_FRAME_LOST CORACQ_VAL_EVENT_TYPE_DATA_OVERFLOW CORACQ_VAL_EVENT_TYPE_EXTERNAL_TRIGGER_IGNORED CORACQ_VAL_EVENT_TYPE_VERTICAL_TIMEOUT
CORACQ_PRM_SIGNAL_STATUS	CORACQ_VAL_SIGNAL_HSYNC_PRESENT CORACQ_VAL_SIGNAL_VSYNC_PRESENT CORACQ_VAL_SIGNAL_PIXEL_CLK_PRESENT
CORACQ_PRM_DETECT_PIXEL_CLK	Not available
CORACQ_PRM_DETECT_HACTIVE	Available
CORACQ_PRM_DETECT_VACTIVE	Available
CORACQ_PRM_FLAT_FIELD_ENABLE *mono only*	True / False
CORACQ_PRM_FLAT_FIELD_SELECT	0
CORACQ_CAP_FLAT_FIELD_OFFSET	min = 0 max = 255 step = 1
CORACQ_CAP_FLAT_FIELD_GAIN	min = 0 max = 255 step = 1
CORACQ_CAP_FLAT_FIELD_GAIN_DIVISOR	0x80
CORACQ_CAP_FLAT_FIELD_PIXEL_REPLACEMENT	TRUE
CORACQ_CAP_SERIAL_PORT_INDEX	Supported

X64-CL iPro Memory Error with Area Scan Frame Buffer Allocation

The memory error message [**Error: "CorXferConnect" <Xfer module> - No memory ()**] may occur when loading a Sapera camera file, or when the application configures a frame buffer for area scan cameras. The problem is that the X64-CL iPro does not have enough onboard memory for the requested frame buffers.

See section "Memory Requirements with Area Scan Acquisitions" on page 27.

X64-CL iPro Sapera Servers & Resources

Servers and Resources

Servers		Resources			
Name	**Description**	**Type**	**Name**	**Index**	**Description**
X64-CL_iPro_1	X64-CL iPro Medium	Acquisition	CamLink Medium Mono	0	CamLink Medium configuration, monochrome output, Camera #1
			CamLink Medium RGB	1	CamLink Medium configuration, RGB output, Camera #1
X64-CL_iPro_1	X64-CL iPro Dual Base	Acquisition	CamLink Base Mono #1	0	CamLink Base configuration, monochrome output, Camera #1
			CamLink Base Mono #2	1	CamLink Base configuration, monochrome output, Camera #2
			CamLink Base Color RGB #1	2	CamLink Base configuration, color output, Camera #1
			CamLink Base Color RGB #2	3	CamLink Base configuration, color output, Camera #2
X64-CL_iPro_1	X64-CL iPro Base	Acquisition	CamLink Bayer Decoder #1	0	CamLink Base configuration, Camera #1, Bayer Decoder
X64-CL_iPro_1	X64-CL iPro\Lite Base	Acquisition	CamLink Base Mono	0	CamLink Base configuration, monochrome output

Transfer Resource Locations

The following table illustrates all possible source/destination pairs in a transfer.

Source	Transfer passing through	Destination
X64-CL iPro Acquisition	1 to 2^{17} internal buffers	1 to 2^{17} Host Buffers
X64-CL iPro Acquisition	1 to 2^{17} internal buffers & the X64 internal processor	1 to 2^{17} Host Buffers

Technical Specifications

X64-CL iPro Board Specifications

X64-CL iPro or iPro\Lite Dimensions

Approximately 8.27 in. (21 cm) wide by 4.2 in. (10.7 cm) high (half length PCI)

Digital Video Input & Controls

Input Type	Camera Link Specifications Rev 1.10 compliant 1 Medium or 2 Base
Common Pixel Formats	
Two Base Configuration Pixel Formats/Tap	3 x 8-bit, 2 x 10-bit, 2 x 12-bit, 1 x 14-bit, 1 x 16-bit, 1 x 24-bit RGB
One Medium Configuration Pixel Formats/Tap	4 x 8-bit, 4 x 10-bit, 4 x 12-bit, , 1 x 14-bit, 1 x 16-bit , 1 x 24-bit RGB, 1 x 30-bit RGB, 1 x 36-bit RGB
Scanning	Progressive, Multi-Tap, Multi-Channel, Four quadrant, Tap reversal
Input LUTs	Yes. See section "X64-CL iPro LUT availability" (page 65) for details.
Resolution *note: these are X64-CL iPro maximums, not Camera Link specifications*	Horizontal Minimum: 8 Pixels per tap Horizontal Maximum: 8-bits/pixel x 256K Pixels/line 16-bits/pixel x 128K Pixels/line 32-bits/pixel x 64K Pixels/line 64-bits/pixel x 32K Pixels/line Vertical Minimum: 1 line Vertical Maximum: up to 16,000,000 lines (for area scan sensors)
Pixel Clock Range	up to 85 MHz
Serial Port	Supports communication speeds 9600 to 115 kbps

Controls	Comprehensive event notification includes end/start-of-field/frame/transfer Dual TTL/LVDS trigger inputs programmable as active high or low (edge or level trigger). Trigger must be of the same type in dual Base mode. Trigger must be a minimum 100ns pulse. Dual independent TTL Strobe outputs Quadrature (AB) shaft-encoder inputs for external web synchronization (maximum frequency for any shaft encoder input is 1 MHz)
Processing *Dependant on user loaded firmware configuration*	**Bayer Mosaic Filter**: Hardware Bayer Engine supports one 8, 10 or 12-bit Bayer camera input. Bayer output format supports 8 or 10-bit RGB/pixel. Zero host CPU utilization for Bayer conversion. **Flat Field Correction** (Shading Correction): Real-time Flat-line and Flat-field correction. Compensates for sensor defects such as FPN, PRNU, defective pixels and variations between pixels due to the light refraction through a lens (Shading effect). Supports two independent monochrome cameras concurrently. PRNU (*Photo Response Non Uniformity*): PRNU is the variation in response between sensor pixels. FPN (*Fixed Pattern Noise*): FPN is the unwanted static variations in response for all pixels in the image.

X64-CL iPro\Lite Board Specifications

Digital Video Input & Controls

Input Type	Camera Link Specifications Rev 1.10 compliant 1 Base
Common Pixel Formats	
Base Configuration Pixel Formats/Tap	3 x 8-bit, 2 x 10-bit, 2 x 12-bit, 1 x 14-bit, 1 x 16-bit, 1 x 24-bit RGB
Scanning	Progressive, Multi-Tap, Multi-Channel, Four quadrant, Tap reversal
Input LUTs	Yes. See section "X64-CL iPro LUT availability" (page 65) for details.
Resolution *note: these are X64-CL iPro maximums, not Camera Link specifications*	Horizontal Minimum: 8 Pixels per tap Horizontal Maximum: 8-bits/pixel x 256K Pixels/line 16-bits/pixel x 128K Pixels/line 32-bits/pixel x 64K Pixels/line 64-bits/pixel x 32K Pixels/line Vertical Minimum: 1 line Vertical Maximum: up to 16,000,000 lines (for area scan sensors)
Pixel Clock Range	up to 85 MHz
Serial Port	Supports communication speeds 9600 to 115 kbps
Controls	Comprehensive event notification includes end/start-of-field/frame/transfer A TTL/LVDS trigger input programmable as active high or low (edge or level trigger). Trigger must be a minimum 100ns pulse. TTL Strobe output Quadrature (AB) shaft-encoder inputs for external web synchronization (maximum frequency for any shaft encoder input is 1 MHz)

Host System Requirements

General System Requirements for the X64-CL iPro Series

- 64-bit PCI-X 66MHz compatible
- 64 bit – 66/33 MHz PCI
- 32 bit – 33 MHz PCI
- 5V or 3.3V PCI slot compatible

Operating System Support

- Windows XP, Windows Vista, and Windows 7

Power Requirements

+5 Volt	2 amp typical	*Note: other internal voltages are derived from +5V*
+12 Volt		*As per camera connected and supplied by X64-CL iPro PC power interface*

Environment

Ambient Temperature:	10° to 50° C (operation) 0° to 70° C (storage)
Relative Humidity:	5% to 90% non-condensing (operating) 0% to 95% (storage)

EMI Certifications

Class B, both FCC and CE

EC & FCC DECLARATION OF CONFORMITY

We : CORECO INC.
7075 Place Robert-Joncas, Suite 142,
St. Laurent, Quebec, Canada H4M 2Z2

Declare under sole legal responsibility that the following products conform to the protection requirements of council directive 89/336 EEC on the approximation of the laws of member states relating to electromagnetic compatibility, as amended by directive 93/68/EEC :

FRAME GRABBER BOARD: X64-CL

The products to which this declaration relates are in conformity with the following relevant harmonised standards, the reference numbers of which have been published in the Official Journal of the European Communities :

EN55022 :1998- Residential, Commercial and Light Industry
EN50204: 1995
EN61000-4: 1995, 1996

Further declare under our sole legal responsibility that the product listed conforms to the code of federal regulations CFR 47 part 15 for a class B product.

St. Laurent, Canada — Location
May 13, 2003 — Date

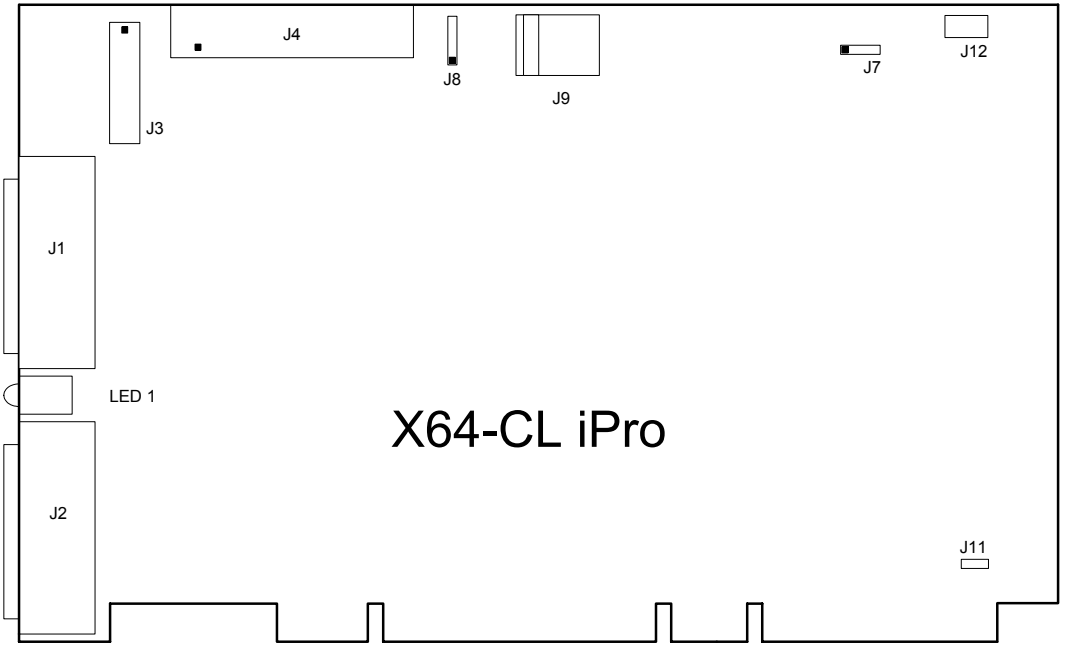
Ghislain Beaupré
Vice-President,
Research & Development

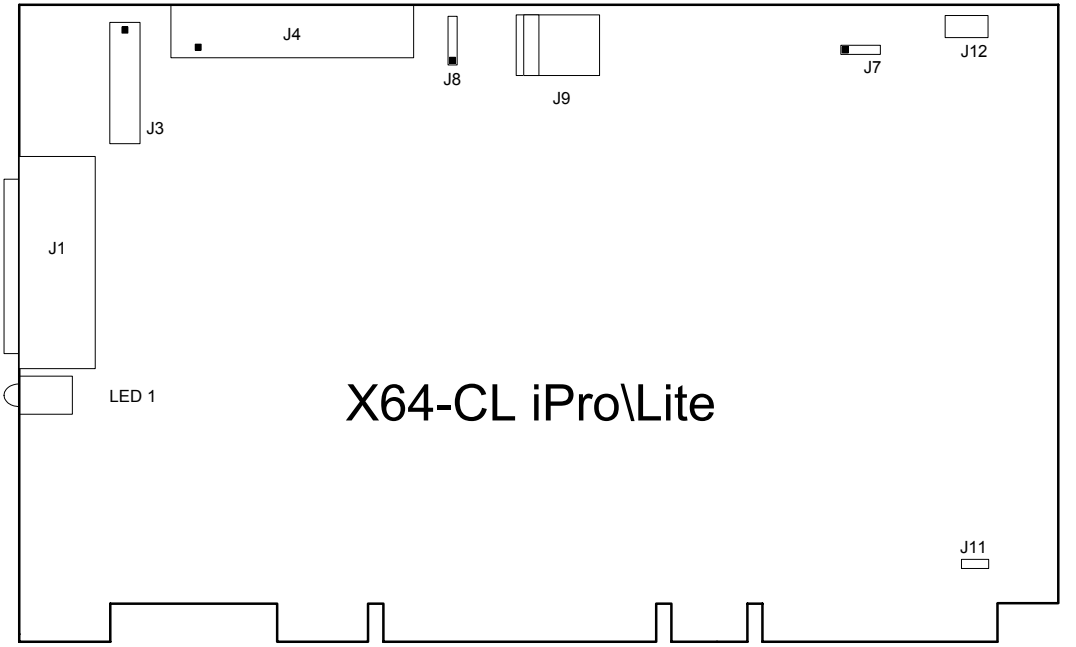
State-of-the-art imaging products

Coreco Imaging 7075 Place Robert-Joncas, Suite 142, Saint-Laurent, Quebec, Canada H4M 2Z2
Telephone: (514) 333-1301 Fax: (514) 333-1388 www.imaging.com

Connector and Switch Locations

X64-CL iPro Board Series Layout Drawings



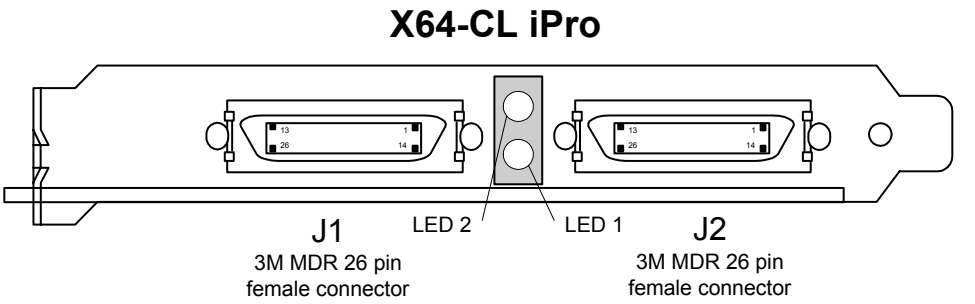


Connector Description List

Connector	Description	Connector	Description
J1	Camera Link Connector	J8	Camera Power Selector
J2	Camera Link Connector	J9	PC power to camera interface.
J3	Reserved	J11	Normal (jumper on) Safe Start Mode (jumper off)
J4	External Signals connector	J12	Reserved
J7	Multi Board Sync		

Connector and Switch Specifications

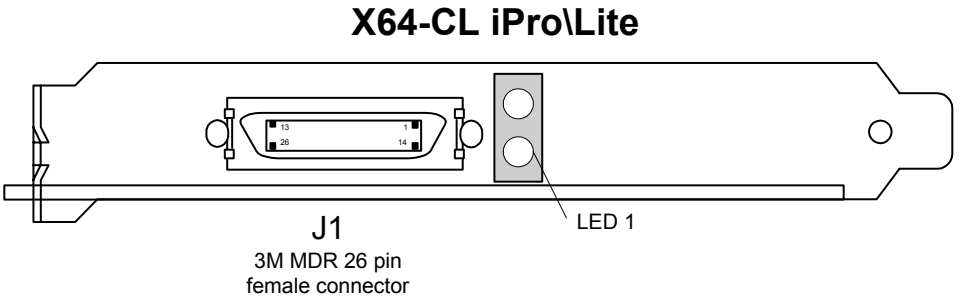
X64-CL iPro End Bracket



The hardware installation process is completed with the connection of a supported camera to the X64-CL iPro board using Camera Link cables (see "Camera Link Cables" on page **107**).

- The X64-CL iPro board supports a camera with one or two Camera Link MDR-26 connectors (two Base or one Medium – see "Data Port Summary" on page 106 for information on Camera Link configurations).
- Connect the camera to the J1 connector with a Camera Link cable. When using a Medium camera, connect the second camera connector to J2.

X64-CL iPro\Lite End Bracket



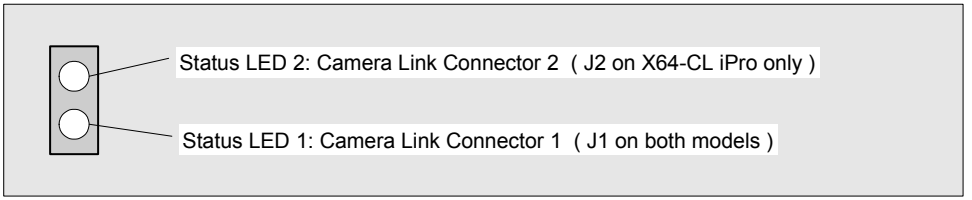
- The X64-CL iPro\Lite supports one Base Camera Link camera.
- Connect the camera to the J1 connector with a Camera Link cable.

Refer to section "Data Port Summary" on page 106 for information on Camera Link configurations

Caution: If the camera is powered by the X64-CL iPro, it is very important that the correct power supply voltage is selected correctly. Refer to "J8: Power to Camera Voltage Selector" on page 102 for information on the selection jumper.

Contact DALSA or browse our web site http://www.imaging.com/camsearch for the latest information on X64-CL iPro supported cameras.

Status LEDs Functional Description



Status LED Modes

- Red: No camera connected or camera has no power.
- Green: Camera connected and is ON. Camera clock detected. No line valid detected.
- Slow Flashing Green: Camera Line Valid signal detected.
- Fast Flashing Green: Acquisition in progress.
- Status LED 2 flashing red, X64-CL iPro board with Medium Configuration only: Camera pixel clock incorrectly connected to J2 instead of J1. (Example - a Base camera is incorrectly connected to J2).

J1: Camera Link Connector 1 (applies to X64-CL iPro and X64-CL iPro\Lite)

Name	Pin #	Type	Description
BASE_X0-	25	Input	Neg. Base Data 0
BASE_X0+	12	Input	Pos. Base Data 0
BASE_X1-	24	Input	Neg. Base Data 1
BASE_X1+	11	Input	Pos. Base Data 1
BASE_X2-	23	Input	Neg. Base Data 2
BASE_X2+	10	Input	Pos. Base Data 2
BASE_X3-	21	Input	Neg. Base Data 3
BASE_X3+	8	Input	Pos. Base Data 3
BASE_XCLK-	22	Input	Neg. Base Clock
BASE_XCLK+	9	Input	Pos. Base Clock
SERTC+	20	Output	Pos. Serial Data to Camera
SERTC-	7	Output	Neg. Serial Data to Camera
SERTFG-	19	Input	Neg. Serial Data to Frame Grabber
SERTFG+	6	Input	Pos. Serial Data to Frame Grabber
CC1-	18	Output	Neg. Camera Control 1
CC1+	5	Output	Pos. Camera Control 1
CC2+	17	Output	Pos. Camera Control 2
CC2-	4	Output	Neg. Camera Control 2
CC3-	16	Output	Neg. Camera Control 3
CC3+	3	Output	Pos. Camera Control 3
CC4+	15	Output	Pos. Camera Control 4
CC4-	2	Output	Neg. Camera Control 4
GND	1, 13, 14, 26		Ground

J2: Camera Link Connector 2 (on X64-CL iPro with Dual Base Configuration)

The Camera Link connector J2 on the X64-CL iPro board is identical to Camera Link connector 1 (J1).

J2: Camera Link Connector 2 (on X64-CL iPro in Medium Configuration)

Name	Pin #	Type	Description
MEDIUM _X0-	25	Input	Neg. Medium Data 0
MEDIUM _X0+	12	Input	Pos. Medium Data 0
MEDIUM _X1-	24	Input	Neg. Medium Data 1
MEDIUM _X1+	11	Input	Pos. Medium Data 1
MEDIUM _X2-	23	Input	Neg. Medium Data 2
MEDIUM _X2+	10	Input	Pos. Medium Data 2
MEDIUM _X3-	21	Input	Neg. Medium Data 3
MEDIUM _X3+	8	Input	Pos. Medium Data 3
MEDIUM _XCLK-	22	Input	Neg. Medium Clock
MEDIUM _XCLK+	9	Input	Pos. Medium Clock
TERM	20		Term Resistor
TERM	7		Term Resistor
	19	Reserved	
	6	Reserved	
	18	Reserved	
	5	Reserved	
	17	Reserved	
	4	Reserved	
	15	Reserved	
	2	Reserved	
	16	Reserved	
	3	Reserved	
GND	1, 13, 14, 26		Ground

Camera Link Camera Control Signal Overview

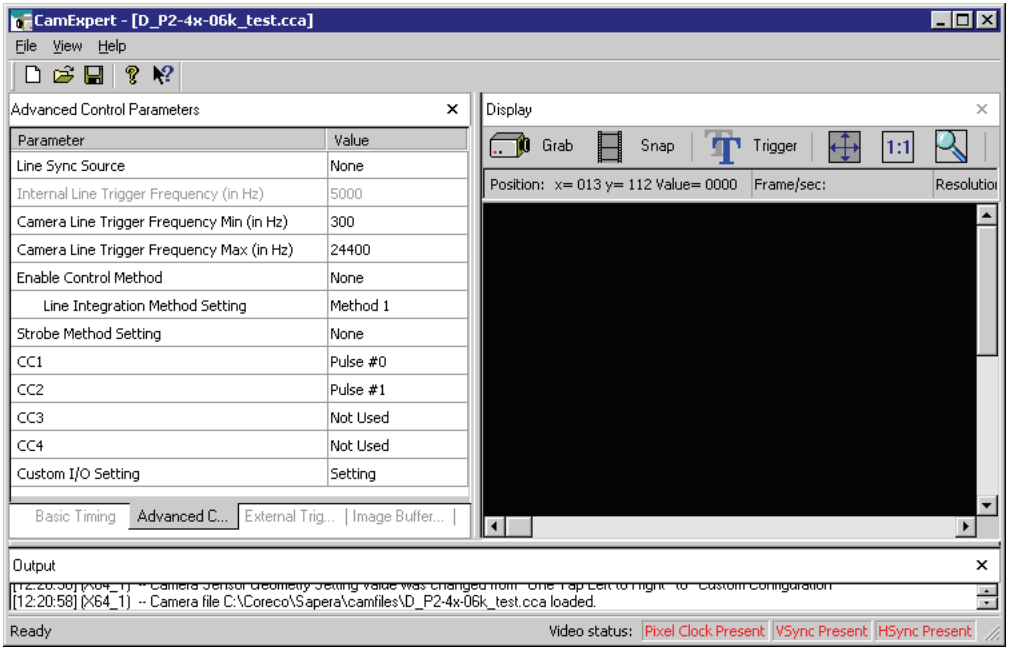
Four LVDS pairs are for general-purpose camera control, defined as camera inputs / frame grabber outputs by the Camera Link Base camera specification. These controls are on J1 (X64-CL iPro and X64-CL iPro\Lite) and also on J2 for the second Base camera input of the X64-CL iPro in two Base configuration.

- Camera Control 1 (CC1)
- Camera Control 2 (CC2)
- Camera Control 3 (CC3)
- Camera Control 4 (CC4)

Each camera manufacture is free to define the signals input on any one or all four control signals. These control signals are used either as camera control pulses or as a static logic state. Control signals not required by the camera are simply assigned as not used. Refer to your camera's user manual for information on what control signals are required.

Note: The X64-CL iPro pulse controller has a minimum resolution of 1 us. When configuring the Camera Link control signals, such as exposure control, etc. use values in increments of 1 us.

The X64-CL iPro can assign any camera control signal to the appropriate Camera Link control. The following screen shot shows the Sapera CamExpert dialog where Camera Link controls are assigned.



J4: External Signals Connector

Warning: Proceed with caution when connecting external devices or other computers to the signals available on J4. Grounds should connect first and devices should be power up at the same time. External signal sources must not have voltage spikes or transients, else damage may occur on the X64-CL iPro.

A recommended solution is to use the optional X-I/O module which provides opto-coupled I/O (see "Appendix: X-I/O Module Option" on page 109). All inputs or outputs provide a more electrically robust interface to outside devices offering better protection against real-world conditions.

J4 Pin Header Numbering Detail

2	4	...	24	26
1	3	...	23	25

J4 Signal Descriptions

Pin #	Signal	X64-CL iPro Medium Description	X64-CL iPro Dual Base Description
1		Reserved	Reserved
2, 4, 6	GND		
3		Reserved	Reserved
5		Reserved	Reserved
7		Reserved	Reserved
8		Reserved	Reserved
9		Reserved	Reserved
10		Reserved	Reserved
11	TrigIn 1 + (*input*)	LVDS Trigger In + or TTL Trigger In (*see note 1*)	*CamLink Base #1* LVDS + or TTL Trigger In Min. pulse—100ns (*see note 2*)
12	TrigIn 1 - (*input*)	LVDS Trigger In - (or TTL Trigger In GND)	*CamLink Base #1* LVDS Trigger - (or TTL Trigger GND)
13	TrigIn 2 + (*input*)	LVDS Trigger In + or TTL Trigger In (*Used for two pulse external trigger with variable frame length linescan acquisition*)	*CamLink Base #2* LVDS + or TTL Trigger In Min. pulse—100ns (*see note 2*)
14	TrigIn 2 - (*input*)	LVDS Trigger In - (or TTL Trigger In GND)	*CamLink Base #2* LVDS Trigger - (or TTL Trigger GND)

15	Phase A + (*input*)	LVDS/RS422 Shaft Encoder phase A + or line trigger used with linescan cameras (*see note 3 & 7*)	LVDS/RS422 Shaft Encoder + or line trigger used with linescan cameras (*see note 3 & 4 & 7*)
16	Phase A - (*input*)	LVDS/RS422 Shaft Encoder phase A - (*see note 7*)	LVDS/RS422 Shaft Encoder - (*see note 7*)
17	Phase B + (*input*)	LVDS/RS422 Shaft Encoder phase B + or line trigger used with linescan cameras (*see note 3 & 7*)	LVDS/RS422 Shaft Encoder + or line trigger used with linescan cameras (*see note 3 & 4 & 7*)
18	Phase B - (*input*)	LVDS/RS422 Shaft Encoder phase B - (*see note 7*)	LVDS/RS422 Shaft Encoder - (*see note 7*)
19	Strobe 2 (*output*)	*not used*	*CamLink Base #2* TTL Strobe Output (*see note 4 & 5*)
20		Reserved	
21	Strobe 1 (*output*)	TTL Strobe Output (*see note 5*)	*CamLink Base #1* TTL Strobe Output (*see note 4 & 5*)
22, 24, 26	GND		
23, 25	DC Power (*see note 6*)	Voltage selected (+12 or +5) via J8 (see "J8: Power to Camera Voltage Selector" on page 102)	Voltage selected (+12 or +5) via J8 (see "J8: Power to Camera Voltage Selector" on page 102)

Notes:

1. **X64-CL iPro :**
   Refer to Sapera parameters CORACQ_PRM_EXT_TRIGGER_LEVEL
   CORACQ_PRM_EXT_FRAME_TRIGGER_LEVEL
   CORACQ_PRM_EXT_TRIGGER_ENABLE
   CORACQ_PRM_EXT_TRIGGER_DETECTION
   See also *.cvi file entries:
   External Trigger Level, External Frame Trigger Level, External Trigger Enable, External Trigger Detection.
2. **X64-CL iPro Dual Base:**
   Sapera parameter CORACQ_PRM_EXT_TRIGGER_LEVEL,
   CORACQ_PRM_EXT_FRAME_TRIGGER_LEVEL is a common control to both Camera Link Base #1 and #2.
   Parameters CORACQ_PRM_EXT_TRIGGER_ENABLE and
   CORACQ_PRM_EXT_TRIGGER_DETECTION are independent for each Camera Link input.
   When detection is CORACQ_VAL_DOUBLE _PULSE_RISING_EDGE or
   CORACQ_VAL_DOUBLE _PULSE_FALLING_EDGE the start trigger is "Trig In 1" and the end trigger is "Trig in 2".
   See also *.cvi file entries:

External Trigger Level, External Frame Trigger Level, External Trigger Enable, External Trigger Detection.

3. **X64-CL iPro:**
   See "Line Trigger Source Selection for Linescan Applications" on page 59 for more information.
   Refer to Sapera parameters CORACQ_PRM_SHAFT_ENCODER_ENABLE
   CORACQ_PRM_SHAFT_ENCODER_DROP
   or refer to CORACQ_PRM_EXT_LINE_TRIGGER_ENABLE
   CORACQ_PRM_EXT_LINE_TRIGGER_DETECTION
   CORACQ_PRM_EXT_LINE_TRIGGER_LEVEL (fixed at LVDS)
   CORACQ_PRM_EXT_LINE_TRIGGER_SOURCE
   See also *.cvi file entries:
   Shaft Encoder Enable, Shaft Encoder Pulse Drop
   or see External Line Trigger Enable, External Line Trigger Detection, External Line Trigger Level, External Line Trigger Source.
4. **X64-CL iPro Dual Base:**
   Parameters are independent for CamLink Base #1 and #2.
5. **X64-CL iPro:**
   Refer to Sapera parameters CORACQ_PRM_STROBE_ENABLE
   CORACQ_PRM_STROBE_POLARITY, CORACQ_PRM_STROBE_LEVEL,
   CORACQ_PRM_STROBE_METHOD, CORACQ_PRM_STROBE_DELAY
   CORACQ_PRM_STROBE_DURATION
   See also *.cvi file entries:
   Strobe Enable, Strobe Polarity, Strobe Level, Strobe Method, Strobe Delay, Strobe Duration.
6. **DC Power Constraints:**
   The supplied host PC voltage is selected (+12 or +5) via the shorting jumper J8 (see "J8: Power to Camera Voltage Selector" on page 102 for details).
   A 1.5A resettable fuse is included on the board. If the fuse is tripped, power off the host computer to reset the fuse.
7. See "Connecting a TTL Shaft Encoder Signal to the LVDS/RS422 Input" on page 99 for details on using a TTL shaft encoder signal.

X64-CL iPro: External Signals Connector Bracket Assembly

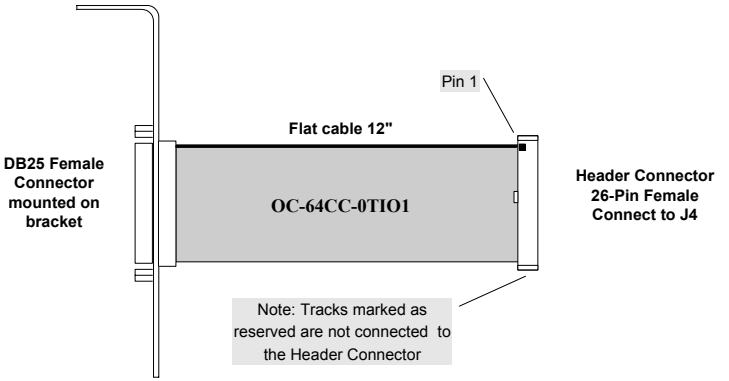
The External Signals bracket (OC-64CC-0TIO1) provides a simple way to bring out the signals from the X64-CL iPro External Signals Connector J4 to a bracket mounted DB25. Install the bracket assembly into an adjacent PC expansion slot and connect the free cable end to the X64-CL iPro J4 header. When connecting to J4, make sure that the cable pin 1 goes to J4 pin 1 (see layout drawings for your board revision: "X64-CL iPro Board Series Layout Drawings" on page 88).

Warning: Proceed with caution when connecting external devices or other computers to the signals available on the External Signals bracket. Grounds should connect first and devices should be power up at the same time. External signal sources must not have voltage spikes or transients, else damage may occur on the X64-CL iPro.

A recommended solution is to use the optional X-I/O module which provides opto-coupled I/O (see "Appendix: X-I/O Module Option" on page 109). All inputs or outputs provide a more electrically robust interface to outside devices offering better protection against real-world conditions.

Note: When using the optional X-I/O module, this external signals cable is not used. Instead a 26 wire cable from the X-I/O J21 connects to X-I/O J4. All external signals described here plus the additional 8 input – 8 output general I/O controls are now on the X-I/O DB37 connector. See "Appendix: X-I/O Module Option" on page 109 for installation and pinout information.

External Signals Connector Bracket Assembly Drawing


Pin 1
Flat cable 12"
DB25 Female Connector mounted on bracket
OC-64CC-0TIO1
Header Connector 26-Pin Female Connect to J4
Note: Tracks marked as reserved are not connected to the Header Connector

External Signals Connector Bracket Assembly Signal Description

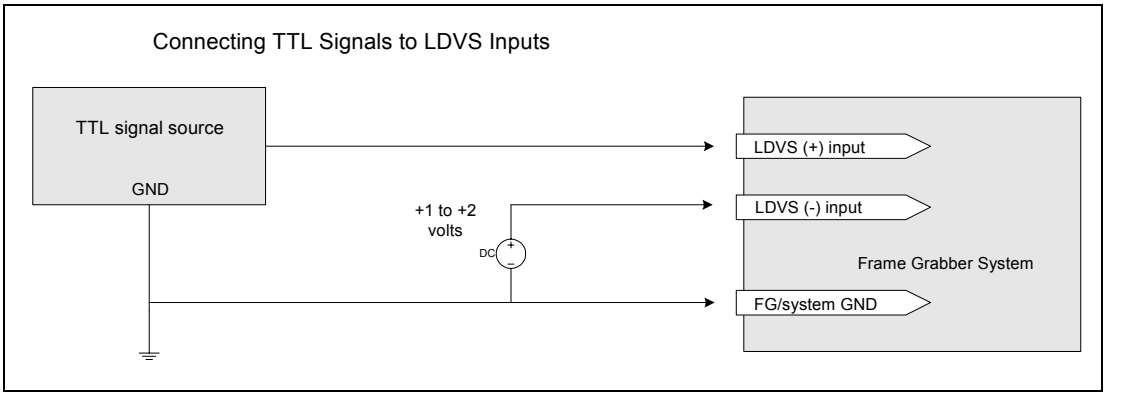
Refer to the table "J4: External Signals Connector " on page 95 for important signal descriptions.

DB25 Pin Number	Signal Names	Connector (to J4)
1, 2, 3, 4, 17, 5, 18, 23	Reserved	1, 3, 5, 7, 8, 9, 10, 20
6	TrigIn 1+	11
19	TrigIn 1-	12
7	TrigIn 2+	13
20	TrigIn 2-	14
8	Shaft Encoder phase A +	15
21	Shaft Encoder phase A -	16
9	Shaft Encoder phase B +	17
22	Shaft Encoder phase B -	18
11	Strobe 1	21
10	Strobe 2	19
12, 13	DC power (+12 or +5 via J8)	23, 25
14, 15, 16, 24, 25,	Ground	2, 4, 6, 22, 24

Connecting a TTL Shaft Encoder Signal to the LVDS/RS422 Input

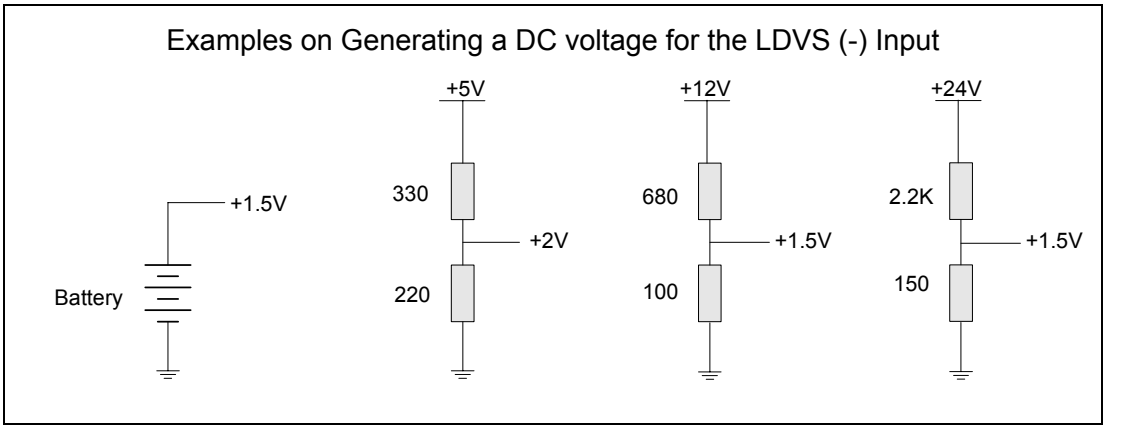
A TTL shaft encoder signal can be directly connected to the X64-CL iPro LVDS/RS422 (+) input but the low side (-) input of the pair must be biased with a DC voltage to ensure reliable operation. This section shows the connection diagram along with suggestions as to how to generate the DC bias voltage. The actual physical wiring is left as an additional detail to interfacing a shaft encoder to the X64-CL iPro to the imaging system.

TTL Shaft Encoder to LVDS/RS422 Input Block Diagram



- LVDS/RS422 (-) input is biased to a DC voltage from +1 to +2 volts.
- This guarantees that the TTL signal connected to the LVDS/RS422 (+) input will be detected as a logic high or low relative to the (-) input.
- The TTL shaft encoder ground, the bias voltage ground, and the X64-CL iPro computer system ground must be connected together.
- The maximum frequency for any shaft encoder input is 1 MHz.

LVDS/RS422 (-) Input Bias Source Generation



- DC voltage for the LVDS/RS422 (-) input can be generated by a resister voltage divider.
- Use a single battery cell if this is more suitable to your system.
- A DC voltage (either +5 or +12) is available on External Signals ConnectorJ4. See "J8: Power to Camera Voltage Selector" on page 102 for information.

External Trigger TTL Input Electrical Specification

The incoming trigger pulse is "debounced" to ensure that no voltage glitch would be detected as a valid trigger pulse. This debounce circuit time constant can be programmed from 0µs to 255µs. Any pulse smaller than the programmed value is blocked and therefore not seen by the acquisition circuitry.

Electrical parameters	Description	Value
TrigIn low	Low logic level input	<= 0.8 V
TrigIn high	High logic level input	>= 2.0 V (max = 5 V)
TrigIn pulse width	Minimum trigger pulse width	100 ns

Sapera parameters for External Trigger:

CORACQ_PRM_EXT_TRIGGER_ENABLE = CORACQ_VAL_EXT_TRIGGER_ON

CORACQ_PRM_EXT_TRIGGER_SOURCE

CORACQ_PRM_EXT_TRIGGER_DETECTION = {CORACQ_VAL_RISING_EDGE, CORACQ_VAL_FALLING_EDGE, CORACQ_VAL_ACTIVE_LOW. CORACQ_VAL_ACTIVE_HIGH}

CORACQ_PRM_EXT_TRIGGER_DURATION: Debounce duration

Strobe TTL Output Electrical Specification

Electrical parameters	Description	Value
$V_{OH\ typ}$	Typical high-level output voltage	3.9V
$I_{OH\ max}$	Maximum high-level output current	-8mA (sourcing)
$I_{OL\ max}$	Maximum low-level output current	8mA (sinking)

Sapera parameters for Strobe :

Refer to Strobe Method in Sapera documentation

CORACQ_PRM_STROBE_ENABLE = TRUE

CORACQ_PRM_STROBE_METHOD={CORACQ_VAL_STROBE_METHOD_1, CORACQ_VAL_STROBE_METHOD_2}

CORACQ_PRM_STROBE_POLARITY={CORACQ_VAL_ACTIVE_LOW, CORACQ_VAL_ACTIVE_HIGH}

CORACQ_PRM_STROBE_DELAY: Pulse offset from trigger event

CORACQ_PRM_STROBE_DELAY_2: Duration of exclusion region

CORACQ_PRM_STROBE_DURATION: Pulse duration

J7: Board Sync

Interconnects multiple X64-CL iPro boards to synchronize acquisitions to one trigger or event. The trigger source can be either an external signal or internal software trigger. The board receiving the trigger is the Master board. while the boards receiving the control signal from the Master board are Slaves.

- **Hardware Connection:** Interconnect two, three, or four X64-CL iPro boards via their J7 connector. The 3 pin cable is wired one to one — i.e. no crossed wires. The cable must be as short as possible and the boards must be in the same system.
- **Master Board Software Setup:** Choose one X64-CL iPro as master. The Sapera parameter CORACQ_PRM_EXT_TRIGGER_SOURCE is set to either *Mode 1–Output to Board Sync* or *Mode 2–Control pulse to Board Sync*. See Sapera documentation for more details.
- **Slave Board Software Setup:** The Sapera parameter CORACQ_PRM_EXT_TRIGGER_SOURCE is set to *From Board Sync*.
- **Test Setup:** The control application starts the acquisition on all slave boards. The acquisition process is now waiting for the control signal from the master board. The master board acquisition is triggered and the acquisition start signal is sent to each slave board (with ~0.8µs delay max).

Contact Technical Support for additional information.

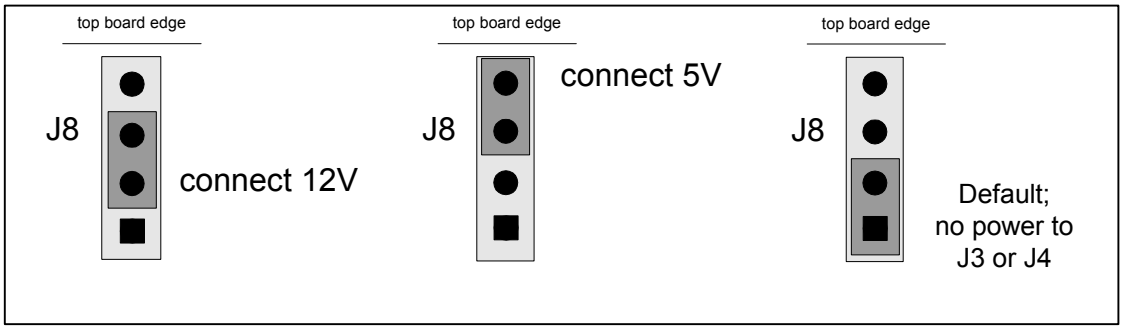
J8: Power to Camera Voltage Selector

When the PC floppy drive power supply cable is connected to J9, a shorting jumper on J8 selects either 5 Vdc or 12 Vdc for the camera power supply. This supply voltage is available on the External Signals Connectorblock.

Important:

For the **X64-CL iPro** a 1.5A power on reset fuse is included on the board. If the fuse is tripped, power off (not just a soft reboot) the computer system to reset the fuse.

J8 on X64-CL iPro



J9: PC Power to Camera Interface

Connect the PC floppy drive power connector to J9 so as to supply DC power to the camera. Place the J8 shorting jumper so as to select 5 Vdc or 12 Vdc for the camera.

J11: Start Mode

- Default Mode: Shunt jumper is installed.
- Safe Mode: Shunt jumper is removed if any problems occurred while updating the X64-CL iPro firmware. With the jumper off, reboot the computer and update the firmware again. When the update is complete, install the jumper and reboot the computer once again. (See "Recovering from a Firmware Update Error" on page 21).

J3, J12: Reserved

Brief Description of Standards RS-232, RS-422, & RS-644 (LVDS)

RS-232

Short for ***r****ecommended* ***s****tandard-232C,* a standard interface approved by the Electronic Industries Association (EIA) connecting serial devices.

The standards for RS-232 and similar interfaces usually restrict RS-232 to 256kbps or less and line lengths of 15M (50 ft) or less.

Transmitted Data (TxD) This signal is active when data is transmitted from the DTE device to the DCE device. When no data is transmitted, the signal is held in the mark condition (logic '1', negative voltage).

Received Data (RxD) This signal is active when the DTE device receives data from the DCE device. When no data is transmitted, the signal is held in the mark condition (logic '1', negative voltage).

DTE (Data Terminal Equipment)
DCE (Data Communication Equipment)

RS-422

RS-422 uses a twisted-pair wire (i.e., 2 wires) for each signal. The differential drive voltage swing is 0 to +5V. RS-422 does not have tri-state capability (its driver is always enabled) and it is therefore usable only in point-to-point communications.

Although RS-422 is noise resistant, due to being differential data can still be damaged by EMI/RFI. A shielded cable can protect the transmitters/receivers from EMI/RFI.

RS-644 (LVDS)

LVDS (Low-Voltage Differential Signaling): method to communicate data using a very low voltage swing (about 350mV) over two differential PCB traces or a balanced cable. LVDS allows single channel data transmission at hundreds of Megabits per second (Mbps).

Camera Link Interface

Camera Link Overview

Camera Link is a communication interface for vision applications developed as an extension of National Semiconductor's Channel Link technology. The advantages of the Camera Link interface are that it provides a standard digital camera connection specification, a standard data communication protocol, and simpler cabling between camera and frame grabber.

The Camera Link interface simplifies the usage of increasingly diverse cameras and high signal speeds without complex custom cabling. For additional information concerning Camera Link, see **http://en.wikipedia.org/wiki/Camera_Link**.

Rights and Trademarks

Note: The following text is extracted from the Camera Link Specification 1.1 (January 2004).

The Automated Imaging Association (AIA), as sponsor of the Camera Link committee, owns the U.S. trademark registration for the Camera Link logo as a certification mark for the mutual benefit of the industry. The AIA will issue a license to any company, member or non-member, to use the Camera Link logo with any products that the company will self-certify to be compliant with the Camera Link standard. Licensed users of the Camera Link logo will not be required to credit the AIA with ownership of the registered mark.

3M™ is a trademark of the 3M Company.

Channel Link™ is a trademark of National Semiconductor.

Flatlink™ is a trademark of Texas Instruments.

Panel Link™ is a trademark of Silicon Image.

Data Port Summary

The Camera Link interface has three configurations. A single Camera Link connection is limited to 28 bits requiring some cameras to have multiple connections or channels. The naming conventions for the three configurations are:

- Base: Single Channel Link interface, single cable connector.
- Medium: Two Channel Link interface, two cable connectors.
- Full: Three Channel Link interface, two cable connectors.

A single Camera Link port is defined as having an 8-bit data word. The "Full" specification supports 8 ports labeled as A to H.

Camera Signal Summary

Video Data

Four enable signals are defined as:

- FVAL Frame Valid (FVAL) is defined HIGH for valid lines.
- LVAL Line Valid (LVAL) is defined HIGH for valid pixels.
- DVAL Data Valid (DVAL) is defined HIGH when data is valid.
- Spare A spare has been defined for future use.

All four enables must be provided by the camera on each Channel Link. All unused data bits must be tied to a known value by the camera.

Camera Controls

Four LVDS pairs are reserved for general-purpose camera control, defined as camera inputs and frame grabber outputs.

- Camera Control 1 (CC1)
- Camera Control 2 (CC2)
- Camera Control 3 (CC3)
- Camera Control 4 (CC4)

Note: the X64-CL iPro by default implements the control lines as follows
(using DALSA Corporation terminology).

(CC1) EXYNC
(CC2) PRIN
(CC3) FORWARD
(CC4) HIGH

Communication

Two LVDS pairs have been allocated for asynchronous serial communication to and from the camera and frame grabber. Cameras and frame grabbers should support at least 9600 baud.

- SerTFG Differential pair with serial communications to the frame grabber.
- SerTC Differential pair with serial communications to the camera.

The serial interface protocol is one start bit, one stop bit, no parity, and no handshaking.

Camera Link Cables

For additional information on Camera Link cables and their specifications, visit the following web sites:

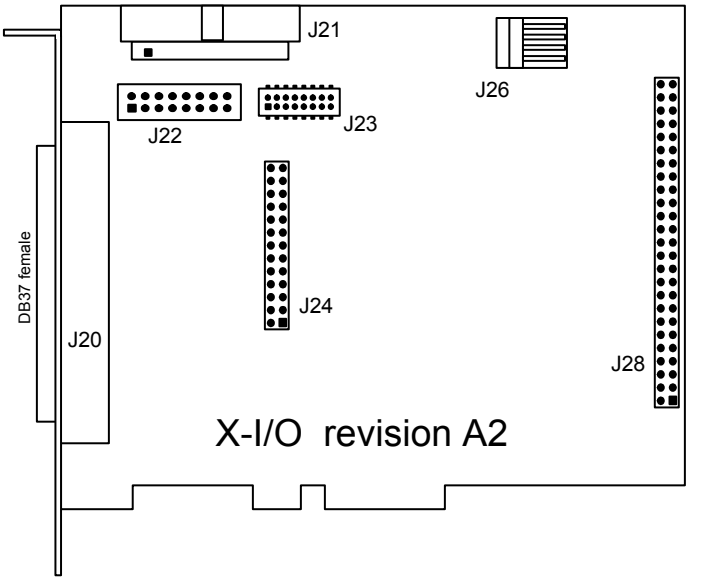
3 M	http://www.3m.com/interconnects *(enter Camera Link as the search keyword)*
Nortech Systems	http://www.nortechsys.com/intercon/CameraLinkMain.htm

Appendix: X-I/O Module Option

X-I/O Module Overview

- The X-I/O module requires X64-CL iPro board driver version 1.10 (or later) and Sapera LT version 5.30 (or later).
- Occupies an adjacent slot to the X64-CL iPro. Slot can be either PCI-32 or PCI-64—no PCI signals or power are used.
- Connects to the X64-CL iPro via a 26 pin flat ribbon cable. J21 on X-I/O to J4 on X64-CL iPro. Note that the external signals connector bracket is removed.
- All X64-CL iPro external signals, such as trigger, shaft encoder, strobe, are available on the X-I/O DB37. See "DB37 Pinout Description" on page 112.
- X-I/O provides 8 outputs software selectable as NPN (current sink) or PNP (source driver) type drivers. See "Outputs in NPN Mode: Electrical Details" on page 113 and "Outputs in PNP Mode: Electrical Details" on page 114.
- X-I/O provides 2 opto-coupled inputs. See "Opto-coupled Input: Electrical Details" on page 115.
- X-I/O provides 6 TTL level inputs with software selectable transition point. See "TTL Input Electrical Details" on page 115.
- X-I/O provides both +5 volt and +12 volt power output pins on the DB37, where power comes directly from the host system power supply.
- Onboard flash memory to store user defined power up I/O states.

X-I/O Module Connector List & Locations



J20	DB37 female external signals connector.
J21	26 pin header connector (interconnect to the X64-CL iPro via supplied ribbon cable).
J22, J23, J24, J28	Reserved.
J26	Connect PC power via floppy drive power cable.

X-I/O Module Installation

Grounding Instructions: Static electricity can damage electronic components. Please discharge any static electrical charge by touching a grounded surface, such as the metal computer chassis, before performing any hardware installation. If you do not feel comfortable performing the installation, please consult a qualified computer technician. **Never** remove or install any hardware component with the computer power on.

Board Installation

Installing an X-I/O Module to an existing X64-CL iPro installation takes only a few minutes. Install the X-I/O board into the host system as follows:

- Power off the computer system that has the installed X64-CL iPro board.
- Disconnect the external signals cable (OC-64CC-0TIO1) if it was used. Remove that cable bracket from the computer.
- Insert the X-I/O module into any free PCI slot (no PCI electrical connections are used), securing the bracket.
- Connect the X-I/O module 26 pin ribbon cable from J21 to the X64-CL iPro board J4.
- Power on the computer again.
- For new X64-CL iPro and X-I/O module installations, simply follow the procedure to install Sapera and the X64-CL iPro driver (start with "Sapera LT Library Installation" on page 10).

X64-CL iPro and X-I/O Driver Update

- If both Sapera 5.30 and X64-CL iPro driver 1.10 need to be installed, follow the procedure "Sapera and Board Driver Upgrades" on page 10. This procedure steps through the upgrade of both Sapera and the board driver—typically required when installing the X-I/O module in the field.
- If the X64-CL iPro installation already has the required Sapera and board driver version, install the X-I/O module and perform a firmware update as described in "Executing the Firmware Loader from the Start Menu" on page 13.

X-I/O Module External Connections to the DB37

Users can assemble their interface cable, using some or all of the signals available on the X-I/O module DB37. Use a male DB37 with thumb screws for a secure fit. Wiring type should meet the needs of the imaging environment.

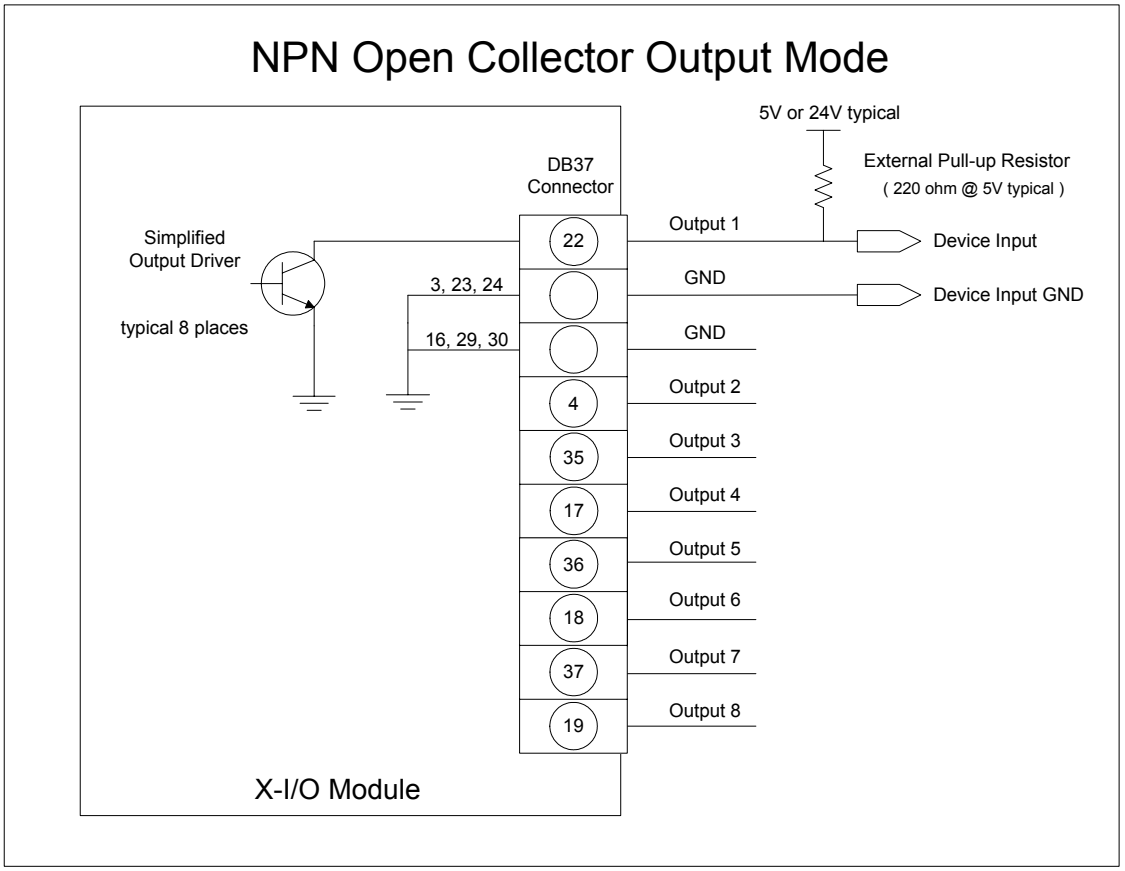
For the external signals Trigger Input, Shaft Encoder Input, and Strobe output, now available on the DB37, refer "J4: External Signals Connector" on page 95 to for signal details.

DB37 Pinout Description

Pin #	Signal	Description
1	IN_OPTO_1+	Input #1 (Opto-coupled)
20	IN_OPTO_1-	
2	IN_OPTO_2+	Input #2 (Opto-coupled)
21	IN_OPTO_2-	
3, 23, 24	Gnd	
22	OUT_TTL_1	output #1
4	OUT_TTL_2	output #2
5	USER_PWR	Power for the TTL Outputs in PNP mode
6	TrigIn 1+	Trigger Input 1 +
25	TrigIn 1-	Trigger Input 1 - (TTL trigger GND)
7	TrigIn 2+	Trigger Input 2 +
26	TrigIn 2-	Trigger Input 2 - (TTL trigger GND)
8	Phase A+	Shaft Encoder Phase A+ (for TTL signals see "Connecting a TTL Shaft Encoder Signal to the LVDS/RS422 Input" on page 99)
27	Phase A-	Shaft Encoder Phase A-
9	Phase B+	Shaft Encoder Phase B+
28	Phase B-	Shaft Encoder Phase B-
10	Strobe 2	TTL Strobe 2 output
11	Strobe 1	TTL Strobe 1 output
16, 29, 30	Gnd	
12	Power	PC +5V (1A max)
31	Power	PC +12V (1A max)
13	IN_TTL_3	Input #3 (TTL)
32	IN_TTL_4	Input #4 (TTL)
14	IN_TTL_5	Input #5 (TTL)
33	IN_TTL_6	Input #6 (TTL)
15	IN_TTL_7	Input #7 (TTL)
34	IN_TTL_8	Input #8 (TTL)
35	OUT_TTL_3	output 3
17	OUT_TTL_4	output 4
36	OUT_TTL_5	output 5
18	OUT_TTL_6	output 6
37	OUT_TTL_7	output 7
19	OUT_TTL_8	output 8

Outputs in NPN Mode: Electrical Details

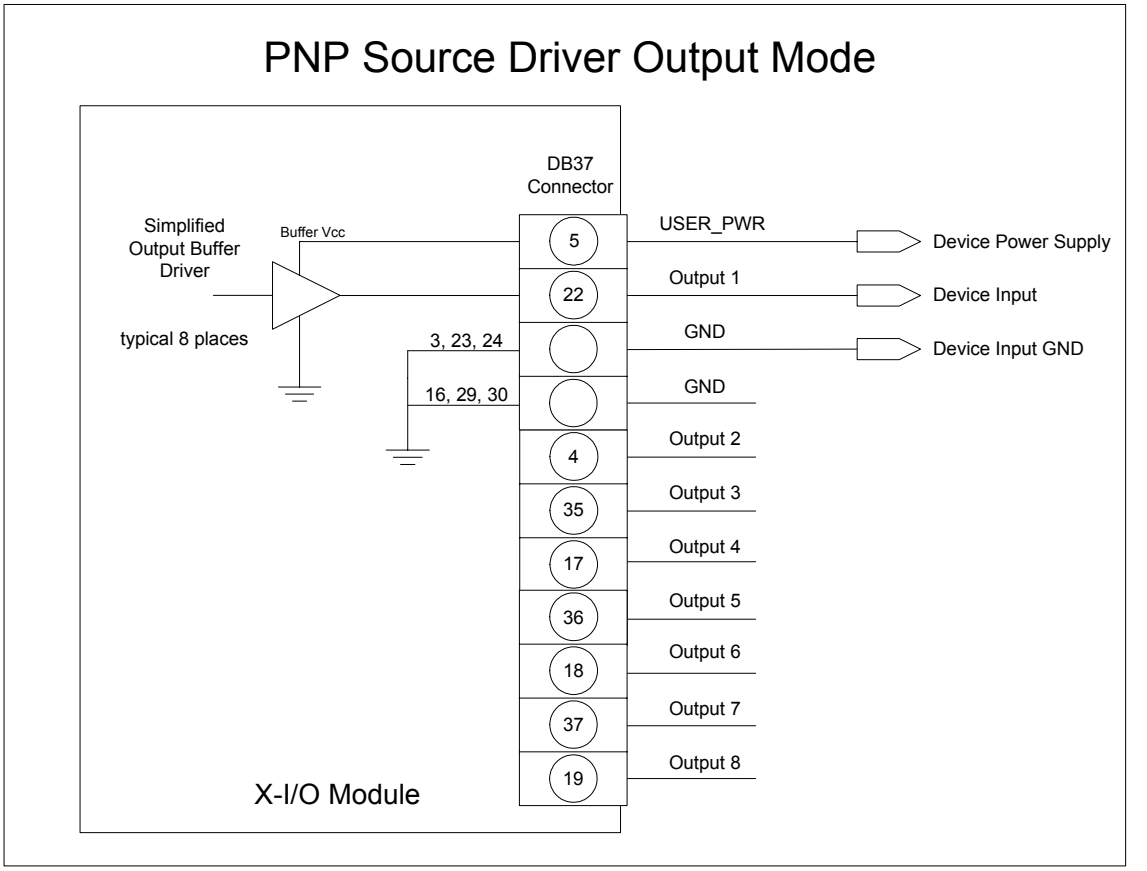
When the outputs are configured for NPN mode (open collector - sink mode) the user is required to provide an external input pull-up resistor on the signal being controlled by the X-I/O output. A simplified schematic and important output specifications follow:



- Each output can sink 700 mA.
- Over-current thermal protection will automatically shut down the output device.

Outputs in PNP Mode: Electrical Details

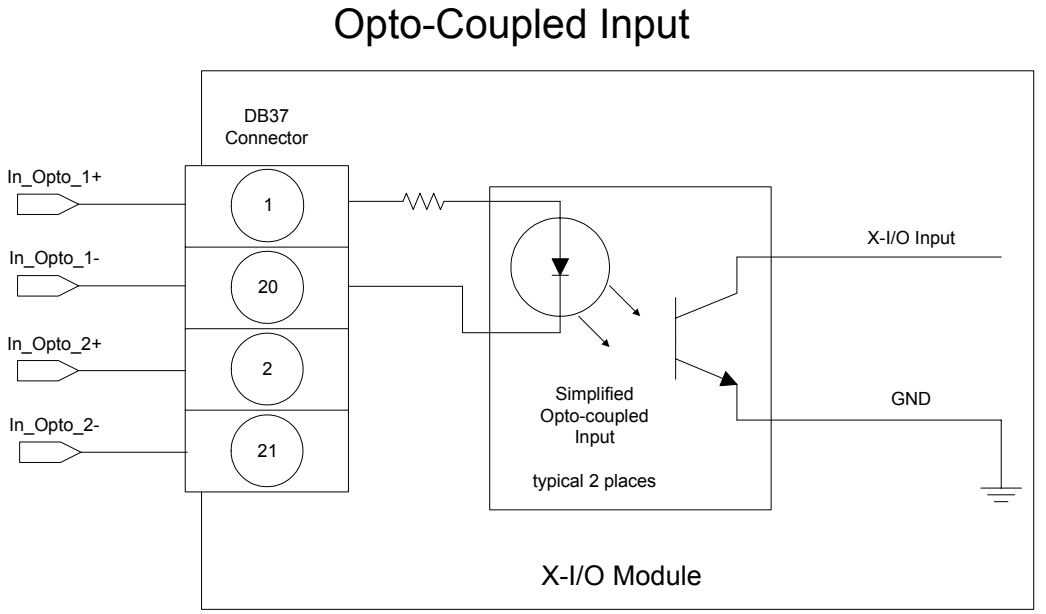
When the outputs are configured for PNP mode (source driver) an external power supply is required to provide the buffer output supply voltage (USER_PWR). A simplified schematic and important output specifications follow:



- User provides the output power supply voltage (7 volts to 35 volts).
- Maximum source driver output current is 350 mA.
- Source driver with over-current protection (all outputs will shut down simultaneously). The over-current fault circuit will protect the device from short-circuits to ground with supply voltages of up to 35V.

Opto-coupled Input: Electrical Details

The two opto-coupled inputs can be used either with TTL or RS422 sources. A simplified input schematic and important electrical specifications are listed below.



Input reverse breakdown voltage	5 volts minimum
Maximum average forward input current	25 mA
Maximum input frequency	200 kHz
Maximum Sapera call-back rate	System processing dependent

TTL Input Electrical Details

The six TTL inputs are software configurable (see "Configuring User Defined Power-up I/O States" on page 116) for standard TTL logic levels or industrial logic systems (typically 24 volts). The design switch points are as follows:

- TTL level mode : trip point at 2V +/- 5%
- Industrial level mode: trip point at 16V +/- 5%

X-I/O Module Sapera Interface

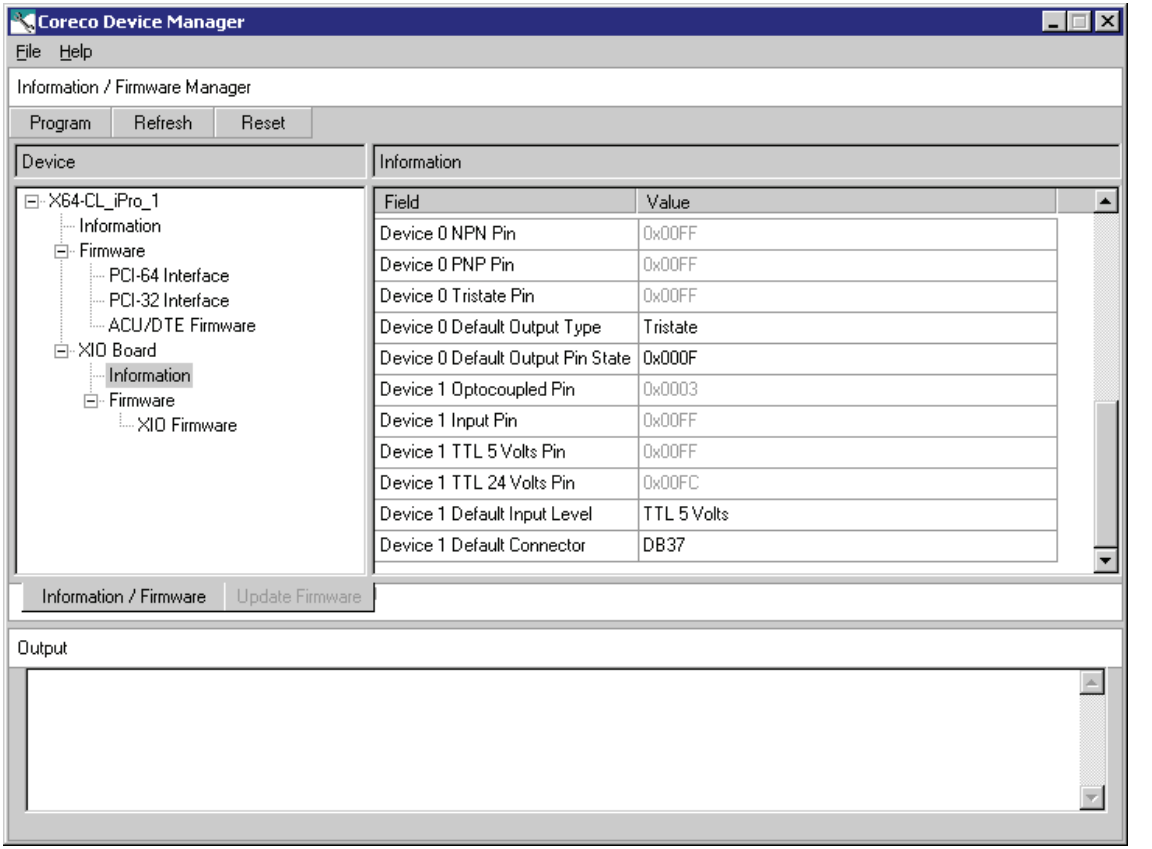
Sapera version 5.30 (or later) provides support for the X-I/O module via an I/O class and demonstration program. Users can use the demonstration program as is, or use the demo program source code to implement X-I/O controls within the custom imaging application.

This section describes configuring the X-I/O module power up state, using the X-I/O demo program, and describes the Sapera Class to program and read the X-I/O module along with sample code.

Configuring User Defined Power-up I/O States

The X-I/O module power up state is stored onboard in flash memory. User configuration of this initial state is performed by the Device Manager program. Run the program via the windows start menu: (**Start • Programs • DALSA • X64-CL iPro Device Driver • Device Manager**).

The Device Manager provides information on the installed X64-CL iPro board and its firmware. With an X-I/O module installed, click on **XIO Board – Information**, as shown in the following figure.



The XIO information screen shows the current status of **Device 0**—the output device, and **Device 1**—the input device. A few items are user configurable for X-I/O board power up state. Click on the item to display a drop list of available capabilities, as described below.

- **Device 0 – Default Output Type**
  choose Tristate mode (i.e. output disconnected), or PNP mode, or NPN mode.
- **Device 0 – Default Output Pin State**
  A window is displayed to select a logic low or high state for each output pin. Click on each pin that should be logic high by default.
- **Device 1 – Default Input Level**
  Select the input logic level as TTL 5 Volts or 24 Volts, dependent on the signal type being input to the X-I/O module.
- **Device 1 – Default Connector**
  DB37 is the supported output connector, as described in this section.

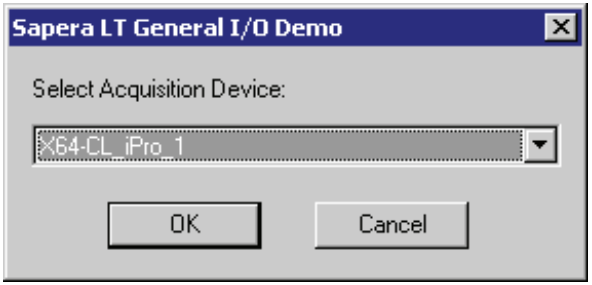
Programming the User Configuration

After changing any user configurable X-I/O mode from the factory default state, click on the **Program** button (located on the upper left), to write the new default state to flash memory. The Device Manager message output window will display "Successfully updated EEProm". The program can now be closed.

Using Sapera LT General I/O Demo

The Sapera General I/O demo program controls the I/O capabilities of the X-I/O module on the Sapera board product. The demo will present to the user only the controls pertaining to the selected hardware (in the case of multiple installed boards).

Run the demo via the windows start menu: (**Start • Programs • DALSA • Sapera LT • Demos • General I/O Demo**). The first menu presents a drop list of all installed Sapera Acquisition Devices with I/O capabilities. In the following figure the X64-CL iPro board is selected. Click OK to continue.



General I/O Module Control Panel

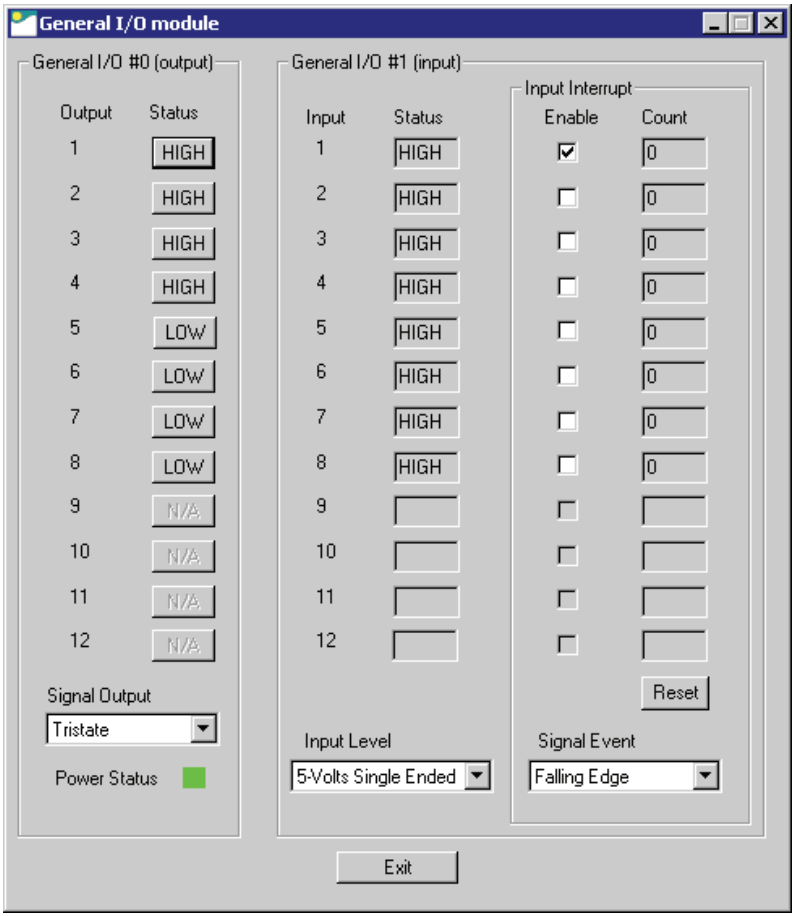
The I/O module control demo presents the I/O capabilities of the installed hardware. The following figure shows the X-I/O module connected to the X64-CL iPro board.

Output Pins: The first column displays the current state of the eight output pins (I/O Device #0).

- The startup default state is user configured using the Device Manager program.
- The state of each output can be changed by clicking on its status button.
- Use the Signal Output drop menu to select the output mode (Tristate, PNP, NPN).

Input Pins: The second section provides input pin status (I/O device #1). Note that this program is a demo, therefore no action takes place on an input event.

- The first column reads the logic level present on each input. The Input Level drop menu changes the logic level from 5V TTL to 24V logic. Use the Device Manager program to select the default logic level type.
- The second column demonstrates activating interrupts on individual inputs. In this demo program, use the Enable box to activate the interrupt on an input. The Count box will tally detected input events. Use the Signal Event drop menu to select which input signal edge to detect. The Reset button clears all event counts.


General I/O module
General I/O #0 (output)
Output
Status
1 HIGH
2 HIGH
3 HIGH
4 HIGH
5 LOW
6 LOW
7 LOW
8 LOW
9 N/A
10 N/A
11 N/A
12 N/A
Signal Output
Tristate
Power Status
General I/O #1 (input)
Input
Status
1 HIGH
2 HIGH
3 HIGH
4 HIGH
5 HIGH
6 HIGH
7 HIGH
8 HIGH
9
10
11
12
Input Level
5-Volts Single Ended
Input Interrupt
Enable
Count
0
Reset
Signal Event
Falling Edge
Exit

Sapera LT General I/O Demo Code Samples

The following source code was extracted from the General I/O demo program. The comments highlight the areas that an application developer needs for embedding X-I/O module controls within the imaging application.

Main I/O Demo code

```
BOOL CGioMainDlg::OnInitDialog()
{

[ . . . ]

// some declarations
UINT32  m_gioCount;
int m_ServerIndex;
int m_ResourceIndex;


// Show the Server Dialog to select the acquisition device
CGioServer dlg(this);
if (dlg.DoModal() == IDOK)
{

    m_ServerIndex = dlg.GetServerIndex();
    m_ServerName = dlg.GetServerName();

    if ( m_ServerIndex != -1)
        {
        // Get the number of resources from SapManager for ResourceGio type by using
        // - the server index chosen in the dialog box
        // - the resource type to enquire for Gio
        m_gioCount=SapManager::GetResourceCount(m_ServerIndex,SapManager::ResourceGio);

        // Create all objects [see the function following]
        if (!CreateObjects()) { EndDialog(TRUE); return FALSE; }

        [ . . . ]

        //Loop for all resources
        for (UINT32 iDevice = 0; (iDevice < MAX_GIO_DEVICE) && (iDevice < m_gioCount);
                                  iDevice++)
        {
            [ . . . ]

            // direct read access to low-level Sapera C library capability to check
            // I/O Output module
            if (m_pGio[iDevice]->IsCapabilityValid(CORGIO_CAP_DIR_OUTPUT))
            status = m_pGio[iDevice]->GetCapability(CORGIO_CAP_DIR_OUTPUT,&capOutput);

            // direct read access to low-level Sapera C library capability to
            // check I/O Input module
            if (m_pGio[iDevice]->IsCapabilityValid(CORGIO_CAP_DIR_INPUT))
```

```
            status = m_pGio[iDevice]->GetCapability(CORGIO_CAP_DIR_INPUT,&capInput);

            [ . . . ]
            // Constructor used for I/O Output module dialog.
            if (capOutput)
            {
            m_pDlgOutput[iDevice] = new CGioOutputDlg(this, iDevice, m_pGio[iDevice]);
            }

            [ . . . ]

            // Constructor used for I/O Input module dialog.
            if (capInput)
            {
            m_pDlgInput[iDevice] = new CGioInputDlg(this, iDevice, m_pGio[iDevice]);
            }
        }   //end for
}   // end if

[ . . . ]
}
```

Function CreateObjects()

```
BOOL CreateObjects()
{
CWaitCursor wait;

// Loop for all I/O resources
for (UINT32 iDevice = 0; (iDevice < MAX_GIO_DEVICE) && (iDevice < m_gioCount);
        iDevice++)
{
    // The SapLocation object specifying the server where the I/O resource is located
    SapLocation location(m_ServerIndex, iDevice);

        // The SapGio constructor is called for each resource found.
        m_pGio[iDevice] = new SapGio(location);

        // Creates all the low-level Sapera resources needed by the I/O object
        if (m_pGio[iDevice] && !*m_pGio[iDevice] && !m_pGio[iDevice]->Create())
        {
            DestroyObjects();
            return FALSE;
        }
    }
return TRUE;
}
```

Output Dialog: CGioOutputDlg class (see Sapera Gui class)

```
void CGioOutputDlg::UpdateIO()
{
    UINT32 output=0;
    UINT32 state=0;
    BOOL status;
    [ . . . ]

    // We loop to get all I/O pins.
    for (UINT32 iIO=0; iIO < (UINT32)m_pGio->GetNumPins(); iIO++)
    {
        [ . . . ]

        // We set the current state of the current I/O pin by using
        // - the pin number on the current I/O resource
        // - the pointer to pin state
        //  ( SapGio ::PinLow if low and SapGio ::PinHigh if high)
        status = m_pGio->SetPinState(iIO, (SapGio::PinState)state);
    }

}
```

Input Dialog: CGioInputDlg class. (see Sapera Gui class)

```
BOOL CGioInputDlg::Update()
{
    SapGio::PinState state = SapGio::PinState::PinLow;
    BOOL status = true;
    UINT32 iIO;
    UINT32 jIO;

    if (m_pGio == NULL)
        return FALSE;

    // We loop to get all I/O pins.
    for (iIO=0; iIO < (UINT32)m_pGio->GetNumPins(); iIO++)
    {
        m_pGio->SetDisplayStatusMode(SapManager::StatusLog, NULL);
        // We get the current state of the current I/O pin by using
        // the pin number on the current I/O resource
        // the pointer to pin state
        //  ( SapGio ::PinLow if low and SapGio ::PinHigh if high)

        status = m_pGio->GetPinState(iIO, &state);
        m_pGio->SetDisplayStatusMode(SapManager::StatusNotify, NULL);

        [ . . . ]
    }

    [ . . . ]
}
```

I/O Event Handling

```
void CGioInputDlg::GioCallbackInfo(SapGioCallbackInfo *pInfo)
{
CGioInputDlg* pInputDlg;
CString strEventCount;

    // We get the application context associated with I/O events
    pInputDlg = (CGioInputDlg*)pInfo->GetContext();

    // We get the current count of I/O events
    strEventCount.Format("%d", pInfo->GetEventCount());

    // We get the I/O pin number that generated an I/O event and apply the changes.
    pInputDlg->m_GioEventCount[pInfo->GetPinNumber()]++;
}
```

DALSA Contact Information

Sales Information

Visit our web site: http://www.imaging.com/

Email: info@dalsa.com

Canada / International

DALSA
7075 Place Robert-Joncas Suite #142
Montreal, Quebec
H4M 2Z2
Canada

Tel: (514) 333-1301
Fax: (514) 333-1388

US Sales Office

DALSA
700 Technology Park Drive
Billerica, Ma. 01821

Tel: (978) 670-2000
Fax: (978) 670-2010

Asia Sales Office

DALSA Asia Pacific
Ikebukuro East 13F
3-4-3 Higashi Ikebukuro,
Toshima-ku, Tokyo
Japan

Tel: +81 3 5960 6353
Fax: +81 3 5960 6354

Technical Support

Any support question or request can be submitted via our web site:

Technical support form via our web page: Support requests for imaging product installations, Support requests for imaging applications	http://www.imaging.com/support
Camera support information	http://www.imaging.com/camsearch
Product literature and driver updates	http://www.imaging.com/download

Glossary of Terms

Bandwidth
Describes the measure of data transfer capacity. PCI devices must share the maximum PCI bus bandwidth when transferring data to and from system memory or other devices.

CAM
Sapera camera file that uses the file extension CCA by default. Files using the CCA extension, also called CAM files (CAMERA files), contain all parameters which describe the camera video signal characteristics and operation modes (i.e. what the camera outputs).

Channel
Camera data path that includes all parts of a video line.

Checksum
A value used to ensure data is stored without error. It is created by calculating the binary values in a block of data using some algorithm and storing the results with the data.

Contiguous memory
A block of physical memory, occupying consecutive addresses.

Firmware
Software such as a board driver that is stored in nonvolatile memory mounted on that board.

Frame buffer
An area of memory used to hold a frame of image data. A frame buffer may exist on the acquisition hardware or be allocated by the acquisition hardware device driver in host system memory.

Grab
Acquiring an image frame by means of a frame grabber.

Host
Refers to the computer system that supports the installed frame grabber.

Host buffer
Refers to a frame buffer allocated in the physical memory of the host computer system.

LSB
Least Significant Bit in a binary data word.

MSB

Most Significant Bit in a binary data word.

PCI 32

Peripheral Component Interconnect. The PCI local bus is a 32-bit high-performance expansion bus intended for interconnecting add-in boards, controllers, and processor/memory systems.

PCI 64

A superset of the PCI specification providing a 64 bit data path and a 66 MHz clock.

Pixel

Picture Element. The number of pixels describes the number of digital samples taken of the analog video signal. The number of pixels per video line by the number of active video lines describes the acquisition image resolution. The binary size of each pixel (i.e., 8-bits, 15-bits, 24-bits) defines the number of gray levels or colors possible for each pixel.

Scatter Gather

Host system memory allocated for frame buffers that is virtually contiguous but physically scattered throughout all available memory.

Tap

Data path from a camera that includes a part of or whole video line. When a camera tap outputs a partial video line, the multiple camera tap data must be constructed by combining the data in the correct order.

VIC

Sapera camera parameter definition file that uses the file extension CVI by default. Files using the CVI extension, also know as VIC files (Configuration Video files), contain all operating parameters related to the frame grabber board (i.e. what the frame grabber can actually do with camera controls or incoming video).

Index

A

Acquisition and Control Unit 67
acquisition bandwidth 29
Acquisition events 67
acquisition module 67
acquisition parameters 53
ACUPlus 6
administrator 9, 10
AUTORUN 10

B

Bayer CFA 5
Bayer Decoder 5
Bayer Filter Decoding 5
Bayer Mosaic Filter 5, 50, 84
Block Diagram 55, 57
BoardInfo.txt 16, 22
boot recovery mode 21
buffer output supply voltage 114

C

cables 17, 90
calibration information 23
camera configuration file 31
camera control 14, 94
Camera file 27, 48, 53, 61, 63
Camera Link 7, 17, 90, 94, 105
Camera Link cabling 14
Camera Link control 94
CamExpert 53, 61, 63
Certifications 87
Color Filter Array 5
communication ports 7
configuration switch 21
connector location 110
Contiguous Memory 19
CORACQ_PRM_EXT_LINE_TRIGGER_DETECTION 97
CORACQ_PRM_EXT_LINE_TRIGGER_ENABLE 97
CORACQ_PRM_EXT_LINE_TRIGGER_LEVEL 97
CORACQ_PRM_EXT_LINE_TRIGGER_SOURCE 97
CORACQ_PRM_EXT_TRIGGER_DETECTION 96
CORACQ_PRM_EXT_TRIGGER_ENABLE 96
CORACQ_PRM_EXT_TRIGGER_LEVEL 96
CORACQ_PRM_SHAFT_ENCODER_DROP 97
CORACQ_PRM_SHAFT_ENCODER_ENABLE 97
CORACQ_PRM_SHAFT_ENCODER_LEVEL 97
CORACQ_PRM_STROBE_DELAY 97
CORACQ_PRM_STROBE_DURATION 97
CORACQ_PRM_STROBE_ENABLE 97
CORACQ_PRM_STROBE_LEVEL 97
CORACQ_PRM_STROBE_METHOD 97
CORACQ_PRM_STROBE_POLARITY 97

D

Data Overflow event 68
Data Transfer Engine 6
debounce circuit time constant 101
debounce control 67
default firmware 65, 66
Device Manager 12, 16, 22, 116
double buffering memory 27
driver upgrade 9

E

End of Frame event 68
End of Transfer event 69
error notification 67
external signals 109
External Signals Connector 60, 61, 62, 91, 95, 99
External Signals Connector Bracket Assembly 61, 98

F

failure - firmware upgrade 21
Firmware Loader 12
firmware revision 16
Fixed Pattern Noise (FPN) 5
Flat Field Correction 5, 48, 84
Found New Hardware Wizard 11
frame buffer 19, 62
Frame Lost event 68
Frame Sync 63
FRAME_RESET 62

H

half length PCI 83

HTML help 2, 3
HyperTerminal 7, 14

I

I/O available capabilities 117
I/O demo program 117
I/O Device 0 117
I/O Device 1 117
I/O flash memory 116
I/O input event 118
I/O input trip points 115
I/O interface cable 111
I/O interrupts 118
I/O NPN output mode 117
I/O output modes 109
I/O PNP output mode 117
I/O power up state 116
I/O sample code 116
I/O source code 120
I/O Tristate output mode 117
image processing 3
Imaging drivers 25
Industrial level mode 115
input logic level 117
input pin status 118
input pull-up resistor 113
Internet 2

J

jumpers 18, 91, 102

L

launch.exe 10
Line Scan 6, 61
Log Viewer program 26
LUT availability 65, 66
LVDS pairs 94

M

maximum data transfer 6
MDR-26 17, 90
memory error 80
multi-board sync 102

N

National Semiconductor 105
NPN mode 113

O

OC-64CC-0TIO1 61
opto-coupled input specs 115
out-of-memory error 19
output sink current 113
output source current 114

P

PCI bus latency 67
PCI Bus Number 24
PCI configuration registers 24
PCI configuration space 22, 24, 29
PCI conflict 22
PDF 2, 3
Phase A 61
Phase B 61
Photo Response Non Uniformity (PRNU) 5
Physical Dimensions 83
Pixel Replacement 5
PNP mode 114
Power Requirements 86
product models 2
programming I/O flash 117

Q

Quadrature-Shaft-Encoder 7

S

Sapera Acquisition Devices 117
Sapera buffers allocation 19
Sapera CamExpert 29
Sapera CD-ROM 9, 10
Sapera configuration program 14, 15, 19
Sapera LT Development Library 10
Sapera LT User's manual 10
Sapera messaging 19
scatter gather buffers 20
Scatter-Gather 6
serial communication port 14
server list 19
Shading Correction 5
shaft encoder 7, 61
software trigger 29
source/destination pairs 82
Static electricity 9, 110

system COM port 14

T

technical support 9, 16, 26, 28
transfer module 68
trigger 7, 61, 62
trigger signal 67
TTL shaft encoder 99

U

user defined I/O state 109
USER_PWR 114

V

viewer program 16, 26
virtual frame buffer 62
visual LED indicators 7

W

Web inspection 61
Windows Event Viewer 22
Windows HyperTerminal 14
Windows Logo testing 11
Windows operating system memory 20
workstation 9, 10

X

X64-CL serial port 14, 15
X64-CL Viewer report 16
X-I/O field installation 111
X-I/O module driver update 111
X-I/O module overview 109